



Sun StorageTek™ Storage Archive Manager (SAM) 歸檔配置與管理指南

版本 4 Update 6

Sun Microsystems, Inc.
www.sun.com

文件號碼 820-1729-10
2007 年 5 月，修訂版 A

請將關於本文件的意見傳送至：<http://www.sun.com/hwdocs/feedback>

Copyright 2007 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. 版權所有。

Sun Microsystems, Inc. 對於本文件所述技術擁有智慧財產權。這些智慧財產權包含 <http://www.sun.com/patents> 上所列的一項或多項美國專利，以及在美國與其他國家/地區擁有的一項或多項其他專利或申請中專利，但並不以此為限。

本文件及相關產品在限制其使用、複製、發行及反編譯的授權下發行。未經 Sun 及其授權人 (如果有) 事先的書面許可，不得使用任何方法、任何形式來複製本產品或文件的任何部分。

協力廠商軟體，包含字型技術，其版權歸 Sun 供應商所有，經授權後使用。

本產品中的某些部分可能源自加州大學授權的 Berkeley BSD 系統的開發成果。UNIX 是在美國和其他國家/地區之註冊商標，已獲得 X/Open Company, Ltd. 專屬授權。

Sun、Sun Microsystems、Sun 標誌、AnswerBook2、docs.sun.com、Solaris、Java 與 Sun StorageTek 是 Sun Microsystems, Inc. 在美國及其他國家/地區的商標或註冊商標。

所有 SPARC 商標都是 SPARC International, Inc. 在美國及其他國家/地區的商標或註冊商標，經授權後使用。凡具有 SPARC 商標的產品都是採用 Sun Microsystems, Inc. 所開發的架構。

Mozilla 是 Netscape Communications Corporation 在美國及其他國家/地區的商標或註冊商標。

OPEN LOOK 與 Sun™ Graphical User Interface (Sun 圖形化使用者介面) 都是由 Sun Microsystems, Inc. 為其使用者及授權者所開發的技術。Sun 感謝 Xerox 公司在研究和開發視覺化或圖形化使用者介面之概念上，為電腦工業所做的開拓性貢獻。Sun 已向 Xerox 公司取得 Xerox 圖形化使用者介面之非獨占性授權，該授權亦適用於使用 OPEN LOOK GUI 並遵守 Sun 書面授權合約的 Sun 公司授權者。

美國政府權利 – 商業用途。政府使用者均應遵守 Sun Microsystems, Inc. 的標準授權合約和 FAR 及其增補文件中的適用條款。

本文件以其「原狀」提供，對任何明示或暗示的條件、陳述或擔保，包括對適銷性、特殊用途的適用性或非侵權性的暗示保證，均不承擔任何責任，除非此免責聲明的適用範圍在法律上無效。



請回收



Adobe PostScript

目錄

- 前言 xix
- 1. 簡介 1
 - 功能 1
 - 儲存裝置 3
 - Sun SAM-Remote 軟體 4
 - 使用 Sun Cluster 的高可用性配置 (HA-SAM) 4
 - File System Manager 5
 - ▼ 要呼叫 File System Manager 5
 - 設定階段作業逾時 6
 - 建立其他管理者和使用者帳號 6
 - ▼ 建立其他帳號 6
 - 指定權限等級 7
 - 為多名使用者建立一個帳號 8
 - 從 File System Manager 管理其他伺服器 9
- 2. 使用自動程式庫和手動載入磁碟機 11
 - 自動程式庫作業 12
 - ▼ 啟動可移除式媒體作業 12
 - ▼ 停止可移除式媒體作業 13

- ▼ 開啓自動程式庫 13
- ▼ 關閉自動程式庫 14
- 將卡匣載入自動程式庫 14
 - ▼ 手動載入卡匣 14
- 從磁碟機中卸載卡匣 15
 - ▼ 手動卸載卡匣 15
- 製作卡匣標籤 15
 - ▼ 製作或重新製作磁帶標籤 15
 - ▼ 製作或重新製作光碟標籤 16
- ▼ 稽核磁碟區 17
- ▼ 稽核直接連結式自動程式庫 17
- 使用清理卡匣 18
 - ▼ 搭配條碼使用清理卡匣 18
 - ▼ 不搭配條碼使用清理卡匣 19
 - ▼ 重設清理週期數目 20
 - ▼ 限制清理週期數 20
 - ▼ 清理磁帶機 21
 - 磁帶機自動清理 21
- ▼ 清除媒體錯誤 22
- ▼ 從磁碟機中移除卡住的卡匣 23
- 目錄檔作業，匯入和匯出卡匣 24
 - 追蹤匯出的媒體 – 記錄器 25
 - 從自動程式庫匯入與匯出 26
 - ▼ 從使用信箱的程式庫匯入卡匣 26
 - ▼ 從使用信箱的程式庫匯出卡匣 27
 - ▼ 自不使用信箱的程式庫匯入卡匣 27
 - ▼ 自不使用信箱的程式庫匯出卡匣 28
- ▼ 啓動載入通知 28

使用具備加密功能的磁碟機	29
手動載入磁碟機作業	30
▼ 載入卡匣	30
▼ 卸載卡匣	30
▼ 檢視程式庫目錄檔	30
3. 歸檔	31
歸檔程序簡介	32
歸檔器監控程序	32
歸檔組	32
歸檔操作	33
步驟 1：辨識要歸檔的檔案	34
步驟 2：組合歸檔請求	36
步驟 3：排定歸檔請求	37
步驟 4：為歸檔請求中的檔案進行歸檔	38
範例預設執行結果	39
歸檔記錄檔案與事件記錄	40
資料驗證	41
關於 archiver.cmd 檔案	42
▼ 建立或修改 archiver.cmd 檔案及散佈您的變更	42
archiver.cmd 檔案	43
archiver.cmd 檔案範例	44
使用歸檔器指令	45
全域歸檔指令	46
archivemeta 指令：控制是否歸檔中介資料	46
archmax 指令：控制歸檔檔案的大小	47
bufsize 指令：設定歸檔器緩衝區大小	48
drives 指令：控制用於歸檔的磁碟機數量	49
examine 指令：控制歸檔掃描	49

- interval 指令：指定歸檔間隔 50
- logfile 指令：指定歸檔器記錄檔 51
- ▼ 備份歸檔器記錄檔 51
- notify 指令：重新命名事件通知程序檔 51
- ovflmin 指令：控制磁碟區溢位 52
- scanlist_squash 指令：控制掃描清單合併 54
- setarchdone 指令：控制 archdone 旗標的設定 54
- wait 指令：延遲歸檔器啟動 54
- 檔案系統指令 55
 - fs 指令：指定檔案系統 55
 - 全域與檔案系統指令 55
- 歸檔組指定指令 56
 - 指定歸檔組 56
 - 檔案時間 *search-criterion*：-access 和 -nftv 57
 - 檔案時間 *search-criterion*：-after 58
 - 檔案大小 *search-criterion*：-minsize 和 -maxsize 58
 - 所有者與群組 *search-criterion*：-user 和 -group 59
 - 使用式樣比對的檔案名稱 *search-criterion*：-name *regex* 59
 - 釋放及分段備份 *file-attributes*：-release 和 -stage 61
 - 歸檔組成員身份衝突 62
- 歸檔備份指令 63
 - 歸檔後釋放磁碟空間：-release 64
 - 延遲釋放磁碟空間：-norelease 64
 - 同時使用 -release 和 -norelease 65
 - 設定歸檔時間 65
 - 自動取消歸檔 65
 - 為中介資料指定多個備份 66

歸檔組備份參數	66
控制歸檔檔案的大小：-archmax	67
設定歸檔器緩衝區大小：-bufsize	68
指定用於歸檔請求的磁帶機數量：-drivemax、-drivemin 及 -drives	68
最大化磁碟區上的空間：-fillvsns	70
指定歸檔緩衝區鎖定：-lock	70
製作離線檔案的歸檔備份：-offline_copy	71
指定回收	71
關聯歸檔：-join path	72
控制取消歸檔	73
控制歸檔檔案的寫入方式：-tapenonstop	73
預留磁碟區：-reserve	74
設定歸檔優先權：-priority	77
排定歸檔：-startage、-startcount 與 -startsize	78
VSN 關聯指令	79
VSN 池指令	81
關於磁碟歸檔	82
配置設定指引	83
磁碟歸檔指令	84
▼ 啟動磁碟歸檔	85
磁碟歸檔範例	87
範例 1	87
範例 2	88
範例 3	89
計畫歸檔作業	90
預覽佇列	90

歸檔器範例 91

範例 1：不具備 archiver.cmd 檔案 91

範例 2：與中介資料分開歸檔的資料檔案 93

範例 3：歸檔至各式媒體的使用者檔案與資料檔案 96

範例 4：歸檔至光學媒體的使用者檔案與資料檔案 100

4. 釋放 103

釋放器程序簡介 104

釋放器概念 105

關於部分釋放和部分分段備份 106

系統管理員選項摘要 107

使用者選項摘要 108

關於 releaser.cmd 檔案 109

指定與時間和大小相關的釋放優先權指令 109

檔案時間 109

檔案大小 111

fs 指令：指定個別檔案系統的指令 112

no_release 和 display_all_candidates 指令：指定除錯 112

min_residence_age 指令：指定最短常駐時間 113

logfile 指令：指定記錄檔 113

research_no_release 指令：限制釋放重新歸檔的檔案 114

list_size 指令：調整釋放器候選者清單的大小 115

指定歸檔組中所有檔案的釋放屬性 115

規劃釋放器作業 116

手動執行釋放器 117

5. 分段備份 119

- 關於 stager.cmd 檔案 119
 - drives 指令：指定磁碟機數量 120
 - 設定分段備份緩衝區大小 121
 - 指定記錄檔 122
 - 指定分段備份請求數目 125
 - stager.cmd 檔案範例 125
 - 指定歸檔組中所有檔案的分段備份屬性 125
- 設定預覽請求的優先權 126
 - 全域 VSN 與時間指令 127
 - 全域標記指令或檔案系統專用的標記指令 127
- 計算預覽請求的總優先權 129
- 設定預覽請求優先機制 129
 - 範例 1：強制分段備份請求 130
 - 範例 2：強制歸檔請求 130
 - 範例 3：依媒體排定請求的優先權 130
 - 範例 4：複合排定優先權 131

6. 回收 133

- 回收程序簡介 133
- 回收磁帶櫃 135
 - 使用回收指令 135
 - 指定記錄檔：logfile 指令 135
 - 防止回收：no_recycle 指令 136
 - 指定回收整個自動程式庫：程式庫指令 136
 - 配置回收器 137
 - 建立 recycler.cmd 檔案 138
 - ▼ 建立 recycler.cmd 檔案 138
 - recycler.cmd 檔案範例 138

- ▼ 執行回收器 140
- 建立回收器的 crontab 檔案 141
- ▼ 移除 `-recycle_ignore` 和 `ignore` 參數 142
- 建立 `recycler.sh` 檔案 142
- 回收磁碟歸檔磁碟區 143
- 編輯 `archiver.cmd` 檔案 143
- ▼ 執行回收器 144
- 建立回收器的 crontab 檔案 144
- ▼ 移除 `-recycle_ignore` 和 `ignore` 參數 144
- 回收以實現歸檔備份保留 145

7. 使用 Sun SAM-Remote 軟體 147

Sun SAM-Remote 軟體簡介 147

- 功能 148
- 需求 149
- 限制 149
- 用戶端與伺服器的互動 150
 - Sun SAM-Remote 伺服器簡介 150
 - Sun SAM-Remote 用戶端簡介 151
 - Sun SAM-Remote 伺服器與 Sun SAM-Remote 用戶端之間的互動 151
 - 程式庫目錄 151
 - 歸檔 152

配置 Sun SAM-Remote 軟體 152

- 配置範例 152
- 配置軟體 153
 - ▼ 登入潛在的伺服器與用戶端主機 153
 - ▼ 確認用戶端與伺服器配置 153
 - ▼ 編輯 `mcf` 檔案 155
 - ▼ 定義 Sun SAM-Remote 用戶端 157

- ▼ 在伺服器的 mcf 檔案中定義 Sun SAM-Remote 伺服器 157
 - ▼ 建立 Sun SAM-Remote 伺服器配置檔案 158
 - ▼ 啟用歸檔 161
- 使用 Sun SAM-Remote 軟體進行回收 164
- 於 Sun SAM-Remote 環境中進行回收 – 方法 1 165
 - 伺服器 sky 的配置檔案 166
 - 用戶端 zeke 的配置檔案 167
 - ▼ 配置回收 – 方法 1 167
 - ▼ 回收 no-data VSN 184
 - ▼ 回收 partially full VSN 186
 - 於 Sun SAM-Remote 環境中進行回收 – 方法 2 188
 - ▼ 配置回收 – 方法 2 188
- 8. 進階主題 189**
- 在 Sun Cluster 環境中配置 SAM-QFS 檔案系統 (HA-SAM) 189
- 開始之前 190
 - 配置指示 191
 - ▼ 驗證 SUNWhasam 資源類型註冊檔案 191
 - ▼ 確認目錄和分段備份程式符號連結 192
 - ▼ 在 Sun Cluster 環境中配置 Sun StorageTek SAM 193
 - 使用 samd hastop 指令 194
 - 搭配 HA-SAM 使用 Sun StorageTek 程式庫 194
- 使用裝置記錄 195
- 何時使用裝置記錄檔 196
 - 啟用裝置記錄檔 196
 - ▼ 使用 samset(1M) 指令以啟動裝置記錄檔 196
 - ▼ 編輯 defaults.conf 檔案以啟動裝置記錄檔 196
- 使用可移除式媒體檔案 197
- ▼ 建立可移除式媒體或磁碟區溢位檔案 198

使用區段檔案	199
歸檔	199
災難復原	199
使用系統錯誤功能報告	200
▼ 啓動 SEF 報告	200
▼ 產生 SEF 報告輸出	201
管理 SEF 記錄檔	203
SEF sysevent 功能	204
▼ 建立 SEF sysevent 處理程式	204
A. 使用供應商指定作業程序的程式庫基本作業	205
ADIC/Grau 自動程式庫	206
▼ 匯入卡匣	206
▼ 匯出卡匣	207
Fujitsu LMF 自動程式庫	207
▼ 匯入卡匣	208
▼ 匯出卡匣	208
IBM 3584 UltraScalable 磁帶櫃	209
匯入卡匣	209
清潔磁碟機	209
分割	209
▼ 移除卡匣	209
IBM 3494 程式庫	210
▼ 匯入卡匣	210
▼ 匯出卡匣	210
Sony 8400 PetaSite 直接連結式自動程式庫	211
▼ 匯入磁帶	211

匯出磁帶 212

▼ 未將信箱插槽用為儲存插槽時匯出磁帶 212

▼ 將信箱插槽用為儲存插槽時匯出磁帶 213

▼ 將卡匣移至不同插槽 213

Sony 網路連結式自動程式庫 214

▼ 匯入卡匣 214

▼ 匯出卡匣 215

StorageTek ACSLS 連結式自動程式庫 215

▼ 匯入磁帶 216

▼ 使用信箱匯出磁帶 216

字彙表 217

索引 227

表

表 1-1	自動程式庫常駐程式	3
表 1-2	File System Manager 權限層級	7
表 2-1	samcmd(1M) load 的引數	14
表 2-2	tplabel(1M) 的引數	16
表 2-3	odlabel(1M) 的引數	16
表 2-4	auditslot(1M) 的引數	17
表 2-5	chmed(1M) 的引數	20
表 2-6	chmed(1M) 的引數	22
表 2-7	auditslot(1M) 的引數	22
表 2-8	chmed(1M) 的引數	24
表 2-9	samexport(1M) 的引數	27
表 3-1	歸檔器記錄檔欄位	40
表 3-2	archiver.cmd 檔案指令單位	43
表 3-3	archmax 指令的引數	47
表 3-4	bufsize 指令的引數	48
表 3-5	drives 指令的引數	49
表 3-6	examine 指令之 <i>method</i> 引數的值	50
表 3-7	ovflmin 指令的引數	52
表 3-8	歸檔組指定指令的引數	56
表 3-9	-access <i>age</i> 後綴	57

表 3-10	-minsize 和 -maxsize <i>size</i> 後綴	58
表 3-11	-release 指令的屬性	61
表 3-12	-stage 指令的屬性	62
表 3-13	歸檔組備份參數的引數	67
表 3-14	-drivemax、-drivemin 及 -drives 指令的引數	68
表 3-15	歸檔組範例分割	69
表 3-16	-offline_copy 指令之 <i>method</i> 引數的值	71
表 3-17	歸檔組表單範例	74
表 3-18	所有者組表單範例	75
表 3-19	檔案系統表單範例	75
表 3-20	歸檔優先權	77
表 3-21	-startage、-startcount 與 -startsize 指令的格式	78
表 3-22	VSN 關聯指令的引數	79
表 3-23	VSN 池指令的引數	81
表 4-1	部分釋放的掛載選項	107
表 4-2	使用者釋放選項	108
表 4-3	釋放用的歸檔組指定值	115
表 5-1	drives 指令的引數	120
表 5-2	bufsize 指令的引數	121
表 5-3	event 引數的值	122
表 5-4	分段備份程式記錄檔欄位	124
表 5-5	可在 archiver.cmd 檔案中使用的分段備份指令	125
表 5-6	標記優先權指令	128
表 5-7	請求優先權範例	130
表 6-1	回收方法和媒體類型	135
表 6-2	no_recycle 指令的引數	136
表 6-3	程式庫指令的 <i>parameter</i> 值	137
表 6-4	歸檔組回收指令	143
表 7-1	samu(1M) R 畫面旗標	163
表 8-1	request(1) 指令的引數	198

表 A-1	import(1M) 指令的引數	206
表 A-2	samexport(1M) 指令的引數	207
表 A-3	import(1M) 指令的引數	208
表 A-4	samexport(1M) 指令的引數	208
表 A-5	move(1M) 指令的引數	212
表 A-6	move(1M) 指令的引數	213
表 A-7	import(1M) 指令的引數	214
表 A-8	samexport(1M) 指令的引數	215
表 A-9	import(1M) 指令的引數	216
表 A-10	samexport(1M) 指令的引數	216

前言

本手冊「Sun StorageTek™ Storage Archive Manager 歸檔配置與管理指南」說明 Sun StorageTek Storage Archive Manager (SAM) 版本 4 Update 6 (4U6) 發行版本所支援的儲存和歸檔管理軟體。Sun StorageTek SAM 軟體會自動將檔案從線上磁碟複製到歸檔媒體。歸檔媒體可以由線上磁碟或可攜式媒體卡匣組成。

下列 Sun Solaris™ 作業系統 (Solaris OS) 平台可支援 Sun StorageTek SAM 4U6 發行版本：

- Solaris 9 update 3 04/03 或更新版本
- Solaris 10

本手冊是針對負責配置和維護 Sun StorageTek SAM 軟體的系統管理員所撰寫的。我們假設您是系統管理員，並知道 Solaris 作業系統管理程序，包含建立帳號、執行系統備份及其他基本的 Solaris 系統管理工作。

備註 – 您可以購買 Sun StorageTek QFS 和 Sun StorageTek SAM 軟體的授權，以合併 Sun StorageTek QFS 檔案系統和 Sun StorageTek SAM 軟體中的儲存與歸檔管理員。這類系統稱為 *SAM-QFS*。

除非必須加以說明，否則本手冊將不會針對 SAM-QFS 配置提供圖解。關於本手冊中的儲存和歸檔管理部分，您可以假設 Sun StorageTek SAM 的參照也適用於 SAM-QFS 配置。同樣地，關於檔案系統設計和功能的部分，您可以假設 Sun StorageTek QFS 的參照也適用於 SAM-QFS 配置。

本書架構

本手冊包含以下各章：

- 第 1 章提供簡介資訊。
- 第 2 章說明基本操作。本章中的資訊適用於大多數自動程式庫與手動載入裝置。
- 第 3 章說明歸檔程序。
- 第 4 章說明釋放程序。
- 第 5 章說明分段備份程序。
- 第 6 章說明回收程序。
- 第 7 章說明如何使用 Sun SAM-Remote 軟體。
- 第 8 章說明 Sun StorageTek SAM 作業中的進階選項。
- 附錄 A 說明如何管理程式庫中的卡匣，並提供了僅適用於該類程式庫的操作說明。

字彙表提供本手冊及其他 Sun StorageTek QFS 和 Sun StorageTek SAM 文件資料中所用的專有名詞之定義。

使用 UNIX 指令

本文件不包括介紹基本的 UNIX[®] 指令和操作程序之資訊，如關閉系統、啓動系統及配置裝置。若需此類資訊，請參閱以下文件：

- 系統隨附的軟體文件
- Solaris 作業系統的相關文件，其 URL 為：
<http://docs.sun.com>

Shell 提示符號

表 P-1 顯示了本手冊所使用的 shell 提示符號。

表 P-1 Shell 提示符號

Shell	提示
C shell	電腦名稱%
C shell 超級使用者	電腦名稱#
Bourne shell 與 Korn shell	\$
Bourne shell 與 Korn shell 超級使用者	#

印刷排版慣例

表 P-2 列出本手冊所使用的印刷排版慣例。

表 P-2 印刷排版慣例

字體或符號	意義	範例
AaBbCc123	指令、檔案及目錄的名稱；螢幕畫面輸出。	請編輯您的 .login 檔案。 請使用 <code>ls -a</code> 列出所有檔案。 % You have mail.
AaBbCc123	您所鍵入的內容 (與螢幕畫面輸出相區別)。	% su Password:
AaBbCc123	新的字彙或術語、要強調的詞。將用實際的名稱或數值取代的指令行變數。	這些被稱為類別選項。 您必須是超級使用者才能執行此操作。 要刪除檔案，請鍵入 rm 檔案名稱。
<i>AaBbCc123</i>	保留未譯的新的字彙或術語、要強調的詞。	應謹慎使用 On Error 指令。
「AaBbCc123」	用於書名及章節名稱。	「Solaris 10 使用者指南」 請參閱第 6 章「資料管理」。

表 P-2 印刷排版慣例 (續)

字體或符號	意義	範例
[]	在語法表示上，括號表示該引數是選用引數。	scmadm [-d sec] [-r n[:n][,n]...][-z]
{ arg arg }	在語法表示上，大括號與直線表示必須指定其中一個引數。	sndradm -b { phost shost }
\	指令行結尾的反斜線 (\) 表示該指令繼續到下一行。	atm90 /dev/md/rdsk/d5 \ /dev/md/rdsk/d1

相關文件

本手冊屬於文件集的一部分，該文件集說明 Sun StorageTek QFS 和 Sun StorageTek SAM 軟體產品的作業。表 P-3 顯示這些產品的完整發行版本 4U6 文件集。

表 P-3 相關文件

書名	文件號碼
「Sun StorageTek QFS 檔案系統配置與管理指南」	820-1719-10
「Sun StorageTek QFS 安裝與升級指南」	820-1724-10
「Sun StorageTek Storage Archive Manager 檔案系統配置與管理指南」	820-1744-10
「Sun StorageTek Storage Archive Manager 安裝與升級指南」	820-1734-10
「Sun StorageTek Storage Archive Manager 疑難排解指南」	820-1739-10
「Sun StorageTek QFS 和 Sun StorageTek Storage Archive Manager 4.6 版本說明」	820-1749-10
「Sun StorageTek QFS Linux 用戶端指南」	820-1754-10

線上存取 Sun 文件

Sun StorageTek QFS 和 Sun StorageTek SAM 軟體發行中包含可以從 Sun 的 Network Storage 文件網站或 `docs.sun.com` 檢視的 PDF 檔案。

從 `docs.sun.com` 存取文件

本網站包含 Solaris 及其他許多 Sun 軟體產品的文件。

1. 請連線至以下 URL：

`http://docs.sun.com`

隨即將出現 `docs.sun.com` 網頁。

2. 透過在搜尋方塊中輸入相應的產品來尋找產品的相關文件。

從 Sun 的 Network Storage 文件網站存取文件

本網站包含有關 Network Storage 產品的文件。

1. 請連線至以下 URL：

http://www.sun.com/products-n-solutions/hardware/docs/Software/Storage_Software

隨即會出現 [Storage Software] 頁面。

2. 按一下 Sun StorageTek SAM 軟體的連結。

協力廠商網站

Sun 對於本文件中所提及之協力廠商網站的使用不承擔任何責任。Sun 對於此類網站或資源中的 (或透過它們所取得的) 任何內容、廣告、產品或其材料不做背書，也不承擔任何責任。對於因使用或依靠此類網站或資源中的 (或透過它們所取得的) 任何內容、產品或服務而造成的或連帶產生的實際或名義上之損壞或損失，Sun 概不負責，也不承擔任何責任。

連絡 Sun 技術支援

如果您在本文件中找不到所需之本產品相關技術問題的解答，請至：

<http://www.sun.com/service/contacting>

授權

如需有關取得 Sun StorageTek SAM 軟體授權的資訊，請連絡您的 Sun 銷售代表，或您的授權服務供應商 (ASP)。

安裝協助

若要取得安裝與配置方面的服務，請連絡 Sun 企業服務 (1-800-USA4SUN)，或連絡當地的企業服務業務代表。

Sun 歡迎您提出寶貴意見

Sun 致力於提高文件品質，因此誠心歡迎您提出意見和建議。請至下列網址提出您對本文件的意見：

<http://docs.sun.com/app/docs/form/comments>

隨函請附上文件書名與文件號碼：「Sun StorageTek SAM 歸檔配置與管理指南」，文件號碼 820-1729-10。

第 1 章

簡介

Sun StorageTek SAM 環境提供具備儲存、歸檔管理和擷取功能的可配置檔案系統。Sun StorageTek SAM 軟體會將線上磁碟快取記憶體中的檔案複製到歸檔媒體，以便歸檔檔案。歸檔媒體可以由其他檔案系統中的磁碟片段組成，或者由自動或手動載入儲存裝置的可移除式磁帶或磁光碟機匣組成。Sun StorageTek SAM 軟體會使用站點指定的使用率臨界值，自動維護線上磁碟。空間。其會釋放與歸檔檔案資料相關的磁碟空間，並在需要時將檔案復原至線上磁碟。

本章提供 Sun StorageTek SAM 元件的技術簡介。本附錄包含以下各節：

- [第 1 頁的「功能」](#)
- [第 3 頁的「儲存裝置」](#)
- [第 4 頁的「Sun SAM-Remote 軟體」](#)
- [第 5 頁的「File System Manager」](#)

功能

Sun StorageTek SAM 儲存與歸檔管理軟體包括伺服器之磁碟快取記憶體中的高性能 UNIX 檔案系統。如需有關此檔案系統的更多資訊，請參閱「Sun StorageTek Storage Archive Manager 檔案系統配置與管理指南」。

位於 Sun StorageTek SAM 環境的其他元件如下所示：

- **歸檔器**，用於自動將線上磁碟快取記憶體中的檔案複製到歸檔媒體。歸檔媒體可由線上磁碟檔案或可移除式媒體卡匣組成。歸檔器預設會在 Sun StorageTek SAM 檔案系統中，自動建立一份所有檔案的歸檔備份，並將其寫入歸檔媒體中。您可以設定歸檔器的組態，以在各種歸檔媒體上建立最多四份歸檔備份。檔案若已分段，便會將每個區段視為一個檔案，並個別進行歸檔。在磁碟檔案與站點可定義的選擇條件組相符後，即初始化歸檔程序。

如需有關歸檔器的更多資訊，請參閱[第 31 頁的「歸檔」](#)。如需有關分段檔案的更多資訊，請參閱[第 199 頁的「使用區段檔案」](#)。

- **釋放器**，可透過釋放合格歸檔檔案所佔用的磁碟區塊，自動維護處於站點指定之百分比使用率臨界值的檔案系統線上磁碟快取記憶體。

釋放是指釋放歸檔檔案資料所使用之主要 (磁碟) 儲存空間的程序。高標記與低標記這兩個臨界值 (均表示為總磁碟空間百分比) 用於管理線上磁碟快取記憶體的可用空間。線上磁碟消耗量超過高標記時，系統會自動開始釋放合格歸檔檔案所佔用的磁碟空間。系統會一直釋放歸檔檔案資料所佔用的磁碟空間，直至達到低標記為止。選取所要釋放的檔案時，會以其大小及存在時間為依據。您可以將檔案的第一部分保留在磁碟上，以加快存取的速度，以及遮罩分段備份延遲。如果檔案已歸檔在區段中，則可以個別地釋放檔案的部分。如需有關釋放器的更多資訊，請參閱第 103 頁的「釋放」。

- **分段備份程式**，用於將檔案資料復原至磁碟快取記憶體。使用者或程序要求已從磁碟快取記憶體釋放的檔案資料時，分段備份程式會自動將檔案資料複製回線上磁碟快取記憶體。

存取已釋放資料區段的檔案時，分段備份程式會自動將檔案或檔案區段資料分段備份回線上磁碟快取記憶體。讀取操作會直接在分段備份作業後進行追蹤，以便在完全分段備份整個檔案之前，應用程式可立即使用該檔案。

Sun StorageTek SAM 軟體程序會自動分段備份請求錯誤。如果傳回分段備份錯誤，系統會嘗試尋找下一個可用的檔案歸檔備份。可以自動處理的分段備份錯誤包括媒體錯誤、無法使用媒體、無法使用自動程式庫及其他錯誤。如需有關分段備份的更多資訊，請參閱第 119 頁的「分段備份」。

- **回收器**，用於清除已過期的歸檔備份之歸檔磁碟區，以使磁碟區可重複使用。

當使用者修改檔案時，即會將歸檔媒體上與這些舊版檔案相關聯的歸檔備份視為過期。系統不再需要這種備份所以能夠將它清除。回收器會識別已過期歸檔備份比例最高的歸檔磁碟區，並將未過期的備份移至個別的磁碟區進行保存。

如果可移除式媒體磁碟區只包含過期的備份，您可以執行以下動作之一：

- 重新標記磁碟區，以立即重複使用。
- 將磁碟區匯出並易地儲存，以做為檔案變更的歷史記錄。您可以使用標準 UNIX 公用程式，從過期的歸檔備份復原之前的檔案版本。

一般使用者不需設定回收程序。如需有關回收的更多資訊，請參閱第 133 頁的「回收」。

儲存裝置

Sun StorageTek SAM 環境支援各種磁帶儲存裝置與磁光裝置。Sun StorageTek SAM 所支援的自動程式庫，可以根據其連接至環境的方式分為以下群組：

- **直接連結。**直接連結式程式庫會使用小型電腦系統介面 (SCSI) 直接連接至主機系統。此方式可稱為直接連結或光纖通道連結。例如，Sun StorageTek 程式庫使用的即是直接連結。Sun StorageTek SAM 系統使用 SCSI 標準直接控制這些自動程式庫。
- **網路連結。**Sun StorageTek SAM 軟體可以配置為程式庫主機系統的用戶端。網路連結式程式庫包括部分的 StorageTek、ADIC/Grau、IBM 及 Sony 程式庫。這些程式庫使用廠商提供的套裝軟體。在這些情況下，Sun StorageTek SAM 軟體會使用針對自動程式庫專門設計的常駐程式與供應商軟體進行介面互動。

表 1-1 將顯示各種自動程式庫特定的常駐程式。

表 1-1 自動程式庫常駐程式

常駐程式	說明
sam-robotsd	監控自動裝置常駐程式的執行。sam-aml1d 常駐程式會自動啟動 sam-robotsd 常駐程式。
sam-genericd	控制直接連結式程式庫與媒體抽換裝置。同時透過 DAS 介面控制 ADIC 程式庫。
sam-stkd	在 ACSAPI 介面中控制 StorageTek 媒體抽換裝置。
sam-ibm3494d	透過 lmcpcd 介面控制 IBM 3494 程式庫。
sam-sonyd	透過 DZC-8000S 介面控制 Sony 網路連結式自動程式庫。

有關支援的儲存裝置清單，請與 Sun Microsystems 銷售代表或授權的服務供應商 (ASP) 連絡。

Sun StorageTek SAM 環境內受管理裝置之間的關係定義在主要配置檔案 `/etc/opt/SUNWsamfs/mcf` 中。mcf 檔案指定 Sun StorageTek SAM 環境中包含的可移除式媒體裝置、程式庫，以及檔案系統。mcf 檔案中為每台設備指定了唯一的設備識別碼。mcf 檔案中的項目亦定義手動掛載的歸檔裝置與自動程式庫目錄檔。

如有必要，系統會使用標準 Solaris 磁碟與磁帶裝置驅動程式。對於 Solaris 作業系統不直接支援的裝置 (如某些磁帶庫與光碟裝置)，Sun StorageTek SAM 套裝軟體中會包括特殊的裝置驅動程式。

Sun SAM-Remote 軟體

Sun SAM-Remote 軟體是一種主從式機制，可以讓使用者在 Sun StorageTek SAM 主機系統之間共用程式庫與其他可移除式媒體裝置。Sun SAM-Remote 軟體可讓您配置多部儲存用戶端，以對中央磁帶櫃或磁光碟程式庫中的檔案進行歸檔及分段備份。例如：如果您的主機系統所在網路跨越廣闊的地理區域，您可以將在某個城市建立的檔案歸檔至遠在異地的程式庫中的卡匣。

如需更多資訊，請參閱第 147 頁的「[使用 Sun SAM-Remote 軟體](#)」。

使用 Sun Cluster 的高可用性配置 (HA-SAM)

透過使用 Sun™ Cluster 軟體，可將 Sun StorageTek SAM 配置為高可用性。Sun Cluster 軟體透過啓用應用程式容錯移轉來提供高可用性。主節點會定期受到監視，且叢集軟體會自動將 Sun StorageTek SAM 功能從失敗的主節點遷移至指定的次要節點。Sun Cluster 軟體藉由允許叢集中的另一節點在主節點失敗時自動託管歸檔工作負荷量，可大量減少當機時間並提升效率。

由於高可用性 SAM (HA-SAM) 依賴於 Sun StorageTek QFS Sun Cluster 代理程式，因此必須在安裝 Sun StorageTek QFS Sun Cluster 代理程式所掛載和管理的共用 Sun StorageTek QFS 檔案系統時附帶安裝此配置。

如需更多資訊，請參閱第 189 頁的「[在 Sun Cluster 環境中配置 SAM-QFS 檔案系統 \(HA-SAM\)](#)」。

File System Manager

File System Manager 軟體是一種基於瀏覽器的圖形化使用者介面，讓您可從中央位置配置和控制一或多個 Sun StorageTek QFS 或 Sun SAM-QFS 伺服器。要存取這個中央位置，您可以使用您的網路上任何主機上的網頁瀏覽器。

此軟體的目的在於提供易於使用的介面，以便於使用者執行與 Sun StorageTek QFS 或 Sun SAM-QFS 伺服器相關的常見作業。為了進一步配置或管理伺服器，請使用伺服器的指令行介面、程序檔、配置檔等等。若需更多資訊和安裝檔案系統管理之指示，請參閱「Sun StorageTek Storage Archive Manager 安裝與升級指南」。

▼ 要呼叫 File System Manager

1. 登入到管理工作站的 Web 伺服器。
2. 在 Web 瀏覽器中，連線至下列 URL：

```
https://hostname:6789
```

鍵入 File System Manager 軟體安裝所在之主機的名稱做為 *hostname*。若您要在主機名稱以外指定網域名稱，請以下列格式指定 *hostname : hostname.domainname*。

注意此 URL 以 https 開頭，而非 http。

隨即會出現 Sun Java™ Web Console 登入畫面。

3. 在 [User Name] 提示符號下，輸入 root 或其他有效的使用者名稱。

備註 – 若是從舊版的 File System Manager 軟體進行升級，則亦可使用 samadmin 使用者帳號。您可以在 [User Name] 欄位中鍵入 samadmin，再鍵入 samadmin 密碼，以取得所有 File System Manager 作業的完整存取權。

4. 在 [Password] 提示符號下輸入密碼。
5. 按一下 [Log In]。
6. 按一下 File System Manager。
您現在已登入到 File System Manager。

設定階段作業逾時

Java Web Console 架構的預設階段作業逾時為 15 分鐘。File System Manager 安裝程式可將階段作業逾時變更為 60 分鐘。您可以將階段作業逾時變更為其他值，但建議該值不要超過 60 分鐘，以維護安全性。

若要變更階段作業逾時值，請在管理工作站上輸入下列指令：

```
/opt/SUNWfsmgr/bin/fsmgr session timeout-in-minutes
```

例如，若要將逾時值變更為 45 分鐘，請鍵入：

```
/opt/SUNWfsmgr/bin/fsmgr session 45
```

建立其他管理者和使用者帳號

在初始 File System Manager 配置之後，您可以隨時建立其他的管理員和 Guest 帳號。這些是屬於管理工作站本端的 Guest 帳號。

移除 File System Manager 軟體時，移除程序檔並不會移除其他以手動方式建立的帳號。

您可以使用以下程序建立其他帳號。

▼ 建立其他帳號

1. 在瀏覽器介面之外，以 root 身份登入管理工作站伺服器。
2. 使用 useradd 和 passwd 指令增加各個使用者。

例如，要增加帳號名稱為 bobsmith 的使用者，請鍵入下列指令：

```
# /usr/sbin/useradd bobsmith  
# /usr/bin/passwd bobsmith
```

每個以此方式增加的使用者帳號，對 File System Manager 功能都有唯讀檢視權限。若要增加其他權限，請參閱第 7 頁的「指定權限等級」一節。

指定權限等級

您可為使用者指定 File System Manager 功能的完整或部分存取權。下表列出您可指定給 File System Manager 使用者的五種權限層級。

表 1-2 File System Manager 權限層級

管理權限層級	說明
<code>com.sun.netstorage.fsmgr.config</code>	使用者有無限制的存取權。
<code>com.sun.netstorage.fsmgr.operator.media</code>	使用者可以增加或移除磁帶庫、增加或移除獨立式磁碟機、保留磁碟區序列名稱 (VSN)、匯入 VSN、載入與卸載 VSN、匯出 VSN 等。
<code>com.sun.netstorage.fsmgr.operator.sam.control</code>	使用者可以啟動、停止或閒置歸檔作業。
<code>com.sun.netstorage.fsmgr.operator.file</code>	使用者可以啟動或停止分段備份，以及復原檔案系統。
<code>com.sun.netstorage.fsmgr.operator.filesystem</code>	使用者可以掛載或卸載檔案系統、編輯掛載選項以及執行檔案系統檢查 (fsck)。

若要為使用者指定完整或部分的配置權限，請將下行增加到 `/etc/user_attr` 檔案中：

```
account-name:::auths=privilege-level
```

account-name 是使用者帳號的名稱，而 *privilege-level* 則是您要為此使用者指定的授權層級。

例如，若要為使用者帳號 `bobsmith` 指定完整的權限 (權限層級 `com.sun.netstorage.fsmgr.config`)，請將下行增加到 `/etc/user_attr` 檔案中：

```
bobsmith:::auths=com.sun.netstorage.fsmgr.config
```

若只要針對分段備份和復原檔案系統 (權限層級 `com.sun.netstorage.fsmgr.operator.file`) 及匯出、匯入和指定 VSN (權限層級 `com.sun.netstorage.operator.media`) 指定 `bobsmith` 權限，請將下行增加到 `/etc/user_attr` 檔案中：

```
bobsmith:::auths=com.sun.netstorage.fsmgr.operator.file,  
com.sun.netstorage.fsmgr.operator.media
```

為多名使用者建立一個帳號

您可建立一個供多名使用者使用的通用 File System Manager 帳號，然後增加一個角色，其具有僅供前述某些使用者存取的權限。

1. 使用 `useradd` 和 `passwd` 指令增加帳號。

例如，要為多名使用者增加名為 `guest` 的使用者帳號，請鍵入下列指令：

```
# /usr/sbin/useradd guest
# /usr/bin/passwd guest
```

2. 使用 `roleadd` 和 `passwd` 指令增加角色。

若要在 `guest` 帳號內建立具有特殊權限的 `admin` 角色，請鍵入下列指令：

```
# /usr/sbin/roleadd admin
# /usr/bin/passwd admin
```

3. 在 `/etc/user_attr` 檔案中指定權限層級。

若要指定可復原與分段備份檔案系統的 `admin` 角色權限，請在 `/etc/user_attr` 檔案中增加下列各行：

```
admin::::auths=com.sun.netstorage.fsmgr.operator.file
guest::::type=normal;roles=admin
```

在此情況下，當使用者以 `guest` 身份登入時，File System Manager 會提示使用者選取 [No Role] 或 [Admin]。如果使用者知道 `Admin` 角色密碼，就可以選取 [Admin]、輸入 `Admin` 的密碼，然後擁有復原與分段備份檔案系統的權限。而其他所有的使用者則必須選取 [No Role]，然後擁有唯讀權限。

因為具有相同權限層級的多名使用者可以同步登入軟體，所以可能發生某個使用者的變更覆寫其他使用者之前變更的風險。為避免發生此種狀況，請開發可進行變更者及通知其他使用者方法的策略。

從 File System Manager 管理其他伺服器

預設會將 File System Manager 設定為管理其所在的伺服器。您也可以使用此軟體管理執行 Sun StorageTek QFS 或 Sun StorageTek SAM 軟體的其他伺服器，但必須先配置這些伺服器，以便於 File System Manager 進行存取。配置這些伺服器之後，您即可透過瀏覽器介面管理其檔案系統、歸檔程序和媒體。

新增伺服器：

1. 在瀏覽器介面之外，使用 `telnet` 指令連線至您要增加的伺服器，並以 `root` 身份登入。
2. 使用 `fsmadm(1M) add` 指令新增管理工作站 (安裝了檔案系統管理員的系統) 到可遠端管理此伺服器的主機清單中。

所有透過此指令增加到清單中的主機，均可從遠端管理此伺服器。

例如：

```
# fsmadm add management-station-name.domain-name
```

若要確認增加管理工作站成功與否，可使用 `fsmadm(1M) list` 指令驗證輸出中列有該管理工作站。

3. 以管理者使用者的身份登入至 **File System Manager** 瀏覽器介面。
4. 從 **[Servers]** 頁面，按一下 **[Add]**。

接著便顯示 **[Add Server]** 視窗。

5. 在 **[Server Name]** 或 **[IP Address]** 欄位中，分別鍵入伺服器的名稱或 IP 位址。
6. 按一下 **[OK]**。

第 2 章

使用自動程式庫和手動載入磁碟機

自動程式庫是一種自動控制裝置，無需操作者手動操作，即可載入或卸載可移除式卡匣。自動程式庫也可稱為媒體抽換裝置、光碟櫃、自動裝置、程式庫或媒體程式庫。

本章將說明在 Sun StorageTek SAM 環境中使用自動程式庫及手動載入之磁碟機的相關事宜。此外，本章還將說明以操作者為導向的載入通知工具，此工具在程式庫中沒有所需的磁碟區時向操作者發出警示。

備註 – Sun StorageTek SAM 軟體可以與許多製造商的自動程式庫互通。如需有關程式庫型號、韌體層級及其他相容性的資訊，請連絡 Sun 客戶支援。

某些自動程式庫的一些作業可能會與本章所述不同。若要判斷對於 Sun StorageTek SAM 環境中的自動程式庫，其供應商是否提供有其他作業指示，請參閱第 205 頁的附錄 A 「使用供應商指定作業程序的程式庫基本作業」。

本章包含以下各節：

- 第 12 頁的「自動程式庫作業」
- 第 30 頁的「手動載入磁碟機作業」

自動程式庫作業

許多基本作業在本質上與所有自動程式庫相同。本節將說明下列事項：

- 第 12 頁的「啓動可移除式媒體作業」
- 第 13 頁的「停止可移除式媒體作業」
- 第 13 頁的「開啓自動程式庫」
- 第 14 頁的「關閉自動程式庫」
- 第 14 頁的「將卡匣載入自動程式庫」
- 第 15 頁的「從磁碟機中卸載卡匣」
- 第 15 頁的「製作卡匣標籤」
- 第 17 頁的「稽核磁碟區」
- 第 17 頁的「稽核直接連結式自動程式庫」
- 第 18 頁的「使用清理卡匣」
- 第 21 頁的「清理磁帶機」
- 第 22 頁的「清除媒體錯誤」
- 第 23 頁的「從磁碟機中移除卡住的卡匣」
- 第 24 頁的「目錄檔作業，匯入和匯出卡匣」
- 第 28 頁的「啓動載入通知」
- 第 29 頁的「使用具備加密功能的磁碟機」

▼ 啓動可移除式媒體作業

通常當掛載 Sun StorageTek SAM 檔案系統時，會開始可移除式媒體作業。

- 要手動啟動可移除式媒體作業而無需掛載任何檔案系統，請輸入 `samd(1M) start` 指令。

```
# samd start
```

如果已在輸入上述指令之前執行可移除式媒體操作，則將產生下列訊息：

```
SAM-FS sam-amld daemon already running
```

如需有關 `samd(1M)` 指令的更多資訊，請參閱「`samd(1M)` 線上手冊」。

▼ 停止可移除式媒體作業

您可以停止可移除式媒體作業，將 Sun StorageTek SAM 系統保持在掛載狀態。例如，如果要手動操作程式庫中的卡匣，您即可執行此操作。當作業重新啟動時，即會重新發出擱置的分段備份，繼續執行歸檔。

- 使用 `samcmd(1M) idle` 和 `samd(1M) stop` 指令。

```
samcmd idle eq
samd stop
```

對於 `eq`，請依據 `mcf` 檔案所定義，輸入要停止之磁碟機的設備序號。若要將多部磁碟機設為閒置，請針對 `mcf` 檔案中所配置的各磁碟機，輸入 `samcmd idle eq` 指令。

您也可以使用 `samu(1M)` 操作者公用程式或 File System Manager 將磁碟機設為閒置。

備註 – 請先將 Sun StorageTek SAM 環境中的磁碟機設為閒置，再輸入 `samd(1M) stop` 指令。這將允許歸檔器、分段備份程式及其他程序完成目前作業，當繼續歸檔、分段備份和其他作業時，`samd(1M) stop` 指令執行失敗會造成未預期的結果。

▼ 開啟自動程式庫

當程式庫處於 `on` 狀態時，其受到 Sun StorageTek SAM 系統的控制，並且可以使用一般作業繼續。當開啟程式庫時，Sun StorageTek SAM 軟體會執行以下動作：

- 查詢裝置內部狀態。它將尋找磁帶位置、是否使用條碼等等。
 - 更新目錄及其他內部結構。
- 使用 `samcmd(1M) on` 指令。

```
samcmd on eq
```

對於 `eq`，請依據 `mcf` 檔案中所定義，指定自動程式庫的設備序號。

您也可以使用 `samu(1M)` 或 File System Manager 執行此作業。

▼ 關閉自動程式庫

將程式庫置於 `off` 狀態會停止 I/O 作業，並且從 Sun StorageTek SAM 控制中移除自動程式庫。卡匣不會自行移動，且自動程式庫中的磁碟機仍會維持在 `on` 的狀態。您可能要關閉自動程式庫以執行以下作業：

- 僅針對此自動程式庫停止 Sun StorageTek SAM 作業。
- 關閉自動程式庫的電源。
- 使用 `samcmd(1M) off` 指令。

```
samcmd off eq
```

對於 `eq`，請依據 `mcf` 檔案所定義，指定要定址之自動程式庫的設備序號。

您也可以使用 `samu(1M)` 或 File System Manager 執行此作業。

將卡匣載入自動程式庫

載入是指卡匣從儲存插槽移至磁碟機、並準備接收資料的過程。當請求對磁碟區序列名稱 (VSN) 進行歸檔或分段備份時，即會自動進行載入。

▼ 手動載入卡匣

- 使用 `samcmd(1M) load` 指令。

即使磁碟機處於 `unavail` 狀態，您也可以使用此指令。此指令具有以下兩種可能的格式：

```
samcmd load eq:slot[:partition]  
samcmd load media-type.vsn
```

表 2-1 samcmd(1M) load 的引數

引數	意義
<code>eq</code>	<code>mcf</code> 檔案中所定義的磁碟機設備序號。
<code>slot</code>	程式庫目錄中所指出的儲存插槽號碼。
<code>media-type</code>	表示媒體類型。如需有效媒體類型清單，請參閱「 <code>mcf(4)</code> 線上手冊」。
<code>partition</code>	磁光碟的一面。分割區必須是 1 或 2。此引數不適用在磁帶卡匣。
<code>vsn</code>	表示指定給磁碟區的磁碟區序列名稱。

您也可以使用 `samu(1M)` 或 `File System Manager` 執行此作業。

在手動載入卡匣時，它一般會載入程式庫中的下一個可用磁碟機中。若要使磁碟機無法用於此目的，請使用 `samu(1M)` 公用程式的 `:unavail` 指令或變更使用 `File System Manager` 的裝置狀態。例如，您可以在災難復原作業或分析磁帶時執行此作業。

備註 – Sun StorageTek SAM 在直接連結式程式庫中不支援使用混用的媒體。如果程式庫經過分割，則每個分割區必須僅包含一種媒體類型。

從磁碟機中卸載卡匣

卸載是指從磁碟機移除卡匣的過程。當不再需要某個磁碟區時，便會自動進行卸載。您也可以手動卸載磁碟機。

▼ 手動卸載卡匣

- 使用 `samcmd(1M) unload` 指令。

即使磁碟機處於 `unavail` 狀態，您也可以使用此指令。請使用下列指令：

```
samcmd unload eq
```

對於 `eq`，請依據 `mcf` 檔案中所定義，指定磁碟機的設備序號。

您也可以使用 `samu(1M)` 或 `File System Manager` 執行此作業。

製作卡匣標籤

標記卡匣的程序取決於您標記的是磁帶還是光碟。以下兩個章節將說明這些程序。



注意 – 製作和重新製作卡匣標籤會使任何軟體無法存取目前卡匣上的資料。只有在確定不需要卡匣中儲存資料的情況下再重新製作卡匣標籤。

▼ 製作或重新製作磁帶標籤

- 若要標記新磁帶，請使用 `tplabel(1M)` 指令。

```
tplabel -new -vsn vsn eq:slot
```

- 若要重新製作現有磁帶標籤，請使用 `tplabel(1M)` 指令。

```
tplabel -old vsn -vsn vsn eq:slot
```

表 2-2 `tplabel(1M)` 的引數

引數	意義
<i>vsn</i>	磁碟區序列名稱。如果是重新標記，則新的 VSN 可以與舊的 VSN 相同。
<i>eq</i>	mcf 檔案中所定義的磁碟機設備序號。
<i>slot</i>	程式庫目錄中所指出的磁帶儲存插槽號碼。此引數不適用於手動載入磁碟機。

輸入標記或重新標記磁帶的指令之後，即會將磁帶載入及定位，然後寫入磁帶標籤。如需有關 `tplabel(1M)` 指令的更多資訊，請參閱「`tplabel(1M)` 線上手冊」。

您也可以使用 File System Manager 執行此作業。

▼ 製作或重新製作光碟標籤

- 若要標記新的光碟，請使用 `odlabel(1M)` 指令。

```
odlabel -new -vsn vsn eq:slot:partition
```

- 若要重新製作現有光碟標籤，請使用 `odlabel(1M)` 指令。

```
odlabel -old vsn -vsn vsn eq:slot:partition
```

表 2-3 `odlabel(1M)` 的引數

引數	意義
<i>vsn</i>	磁碟區序列名稱。如果是重新標記，則新的 VSN 可以與舊的 VSN 相同。
<i>eq</i>	mcf 檔案中所定義的磁碟機設備序號。
<i>slot</i>	程式庫目錄中所指出的磁碟儲存插槽號碼。此引數不適用於手動載入磁碟機。
<i>partition</i>	磁光碟的一面。分割區的值必須是 1 或 2。此引數不適用於磁帶卡匣。

輸入標記或重新標記光碟的指令之後，即會將光碟載入及定位，然後寫入光碟標籤。如需有關 `odlabel(1M)` 指令的更多資訊，請參閱「`odlabel(1M)` 線上手冊」。

您也可以使用 File System Manager 執行此作業。

▼ 稽核磁碟區

偶爾，可能需要在程式庫目錄中更新所報告的磁帶或光碟上的剩餘空間。auditslot(1M) 指令將載入包含磁碟區的卡匣、讀取標籤以及更新插槽的程式庫目錄項目。

- 使用 auditslot(1M) 指令稽核磁碟區。

```
auditslot [-e] eq:slot[:partition]
```

表 2-4 auditslot(1M) 的引數

引數	意義
-e	如果指定 -e 選項並且媒體是磁帶，便會更新剩餘空間。否則，其不會變更。
eq	mcf 檔案中所定義的磁碟機設備序號。
slot	程式庫目錄中所指出的儲存插槽號碼。此引數不適用於手動載入磁碟機。
partition	磁光碟的一面。分割區的值必須是 1 或 2。此引數不適用於磁帶卡匣。

如需有關 auditslot(1M) 指令的更多資訊，請參閱「auditslot(1M) 線上手冊」。

您也可以使用 samu(1M) 公用程式的 :audit 指令或使用 File System Manager 執行此作業。

▼ 稽核直接連結式自動程式庫

備註 – 此作業無法在網路連結式自動程式庫中執行。

一個完整的稽核會將每個卡匣載入磁碟機、讀取標籤並更新程式庫目錄。在以下情況應對程式庫進行稽核：

- 在不使用 Sun StorageTek SAM 指令的情況下移動自動程式庫中的卡匣後
 - 懷疑程式庫目錄的狀態並想對其進行更新 (例如停電之後)
 - 在沒有電子信箱的自動程式庫中增加、移除或移動了卡匣
- 使用 samcmd(1M) audit 指令可對直接連結式自動程式庫執行完整稽核。

```
samcmd audit eq
```

對於 eq，請依據 mcf 檔案中所定義，指定自動程式庫的設備序號。

您也可以使用 samu(1M) 公用程式的 :audit 指令或使用 File System Manager 執行此作業。

使用清理卡匣

如果硬體支援清理磁帶功能，即可在 Sun StorageTek SAM 環境中使用清理磁帶。如果磁碟機要求進行清理，系統將自動載入清理磁帶。

如果系統使用已製作條碼的標籤，清理磁帶條碼標籤中的 VSN 必須為 CLEAN 或以 CLN 字母開頭。另外，您也可以使用 `chmed(1M)` 指令將 VSN 標記為清理磁帶，並設定清理週期數目。一個系統中可以載入多個清理磁帶。

清理方式隨製造商而異。如需判別您的設備是否需執行特定的程序，請參閱第 205 頁的「[使用供應商指定作業程序的程式庫基本作業](#)」。

▼ 搭配條碼使用清理卡匣

如果清理卡匣已製作條碼，您可以使用 `import(1M)` 指令將其匯入。

1. 請確定清理卡匣有 CLEAN 條碼，或是以 CLN 字母開頭。
2. 使用 `import(1M)` 指令將清理卡匣匯入自動程式庫中。

```
import eq
```

對於 `eq`，請依據 `mcf` 檔案中所定義，指定自動程式庫的設備序號。

例如，以下指令可將清理磁帶匯入 `mcf` 檔案中編號為 50 的自動程式庫：

```
# import 50
```

您也可以使用 `samu(1M)` 或 File System Manager 執行此作業。

Sun StorageTek SAM 系統會將卡匣從信箱移至儲存插槽，並更新程式庫目錄。此外，在輸入此指令後會設定清理媒體旗標，並且存取數目將根據媒體類型設定為適當的清理週期數目。每使用卡匣清理磁碟機一次，存取數目便會隨之減少一次。

▼ 不搭配條碼使用清理卡匣

如果卡匣未製作條碼，則必須先將其匯入。其不會自動標記為清理卡匣。

1. 使用 `import(1M)` 指令將卡匣匯入自動程式庫中。

```
import eq
```

對於 `eq`，請依據 `mcf` 檔案中所定義，指定自動程式庫的設備序號。

2. 使用 `chmed(1M)` 指令將類型變更為清理卡匣。

```
chmed +C eq:slot
```

對於 `eq`，請指定自動程式庫的設備序號。對於 `slot`，請指定載入清理卡匣的插槽。
在以下範例中，自動程式庫的設備序號為 50，清理卡匣則位於插槽 77：

```
# chmed +C 50:77
```

3. 再次使用 `chmed(1M)` 指令設定清理週期數。

```
chmed -count count-number eq:slot
```

對於 `eq`，請指定自動程式庫的設備序號。對於 `slot`，請指定載入清理卡匣的插槽。
以下指令範例將設定上述步驟中使用的卡匣上的數目：

```
# chmed -count 20 50:77
```

如需有關 `chmed(1M)` 指令的更多資訊，請參閱「`chmed(1M)` 線上手冊」。

▼ 重設清理週期數目

僅對於有限的清理週期數目，清理磁帶才非常有用。您可以使用 `samu(1M)` 公用程式的 `:v` 畫面或 `File System Manager` 檢視剩餘週期數。

`Sun StorageTek SAM` 系統會追蹤每盤清理磁帶所使用的清理週期數，並在剩餘週期數為零時退出磁帶。例如，`DLT` 清理磁帶具有 20 個週期，而 `Exabyte` 清理磁帶具有 10 個週期。每次匯入清理磁帶後，清理週期將重設為該類磁帶的最高週期數目。

如果您的系統可進行自動清理，但自動程式庫中所有清理磁帶的週期數目均為零，則磁碟機將設定為關閉且在 `Sun StorageTek SAM` 錄檔中發出一則訊息。

- 使用 `chmed(1M)` 指令將清理磁帶計數重設為零。

```
chmed -count count media-type.vsn
```

表 2-5 `chmed(1M)` 的引數

引數	意義
<code>count</code>	要重設的清理磁帶之清理週期數目。
<code>media-type</code>	表示媒體類型。如需有效媒體類型清單，請參閱「 <code>mcf(4)</code> 線上手冊」。
<code>vsn</code>	指定給磁碟區的磁碟區序列名稱 (VSN)。

▼ 限制清理週期數

某些磁碟機錯誤可能會導致清理卡匣重複載入，直到所有清理週期耗盡為止。

- 使用 `chmed(1M)` 指令限制清理卡匣的清理週期數。

```
chmed -count count-number eq:slot
```

對於 `eq`，請指定自動程式庫的設備序號。對於 `slot`，請指定載入清理卡匣的插槽。

例如，下列指令將會針對設備序號為 50 的自動程式庫，將其插槽 77 之清理卡匣的清理週期數設定為 20。

```
# chmed -count 20 50:77
```

▼ 清理磁帶機

當無法使用自動清理並且系統使用條碼時，請執行以下程序手動要求清理磁碟機：

- 使用 `cleandrive(1M)` 指令。

```
cleandrive eq
```

對於 `eq`，請依據 `mcf` 檔案中所定義，指定自動程式庫的設備序號。此磁碟機是要與清理卡匣一起載入的磁碟機。

磁帶機自動清理

從 Sun StorageTek SAM 4U4 開始，軟體啟動的磁帶機清理功能的預設設定為 `off`，此設定將強制使用者選擇硬體啟動磁帶機清理策略或軟體啟動磁帶機清理策略。

由硬體啟動的清理策略使用媒體抽換裝置的內建自動清理功能。

- 若要使用此功能，可能須將清理卡匣置入特定的插槽中。
- 請參閱製造商的文件以取得相關指示。

軟體啟動的清理策略使用的是 Sun StorageTek SAM 自動清理功能。

- `logsense` 選項為此功能的增強功能，可避免磁碟機使用過期的清理媒體。
- 若要啟用 Sun StorageTek SAM 自動清理功能，應停用硬體啟動的清理功能，並且在 `defaults.conf` 檔案中設定下行內容：

```
tapeclean = all autoclean on logsense on
```

若要呼叫舊的 Sun StorageTek SAM 自動清理功能 (此功能僅在需要取得磁碟機清理狀態時使用感應資料)，請在 `defaults.conf` 檔案中插入下行：

```
tapeclean = all autoclean on logsense off
```

備註 – 當具有兩個以上磁碟機的程式庫使用自動清理功能時，建議您在每個 Sun StorageTek SAM 目錄中都至少有兩個清理卡匣。如果有需要清理的磁碟機，但是沒有足夠的清理卡匣，那些磁碟機會被放置到 `DOWN` 狀態。

▼ 清除媒體錯誤

當卡匣發生硬體或軟體錯誤時，Sun StorageTek SAM 系統會在 VSN 目錄檔中設定 `media error` 旗標。您可以在任何產生 `media error` 信號的卡匣上使用 `chmed(1M)` 指令清除錯誤，然後再嘗試使用該卡匣。`media error` 旗標會顯示在 `samu(1M)` 公用程式的 `v` 畫面以及 `File System Manager` 中。

備註 – 移除錯誤旗標可能會造成問題。如果您不確定導致錯誤的原因以及是否可安全移除此旗標，請先連絡 Sun 技術支援。

1. 使用 `chmed(1M)` 指令清除 `media error` 旗標。

```
chmed -E media-type.vsn
```

表 2-6 `chmed(1M)` 的引數

引數	意義
<code>media-type</code>	表示媒體類型。如需有效媒體類型清單，請參閱「 <code>mcf(4)</code> 線上手冊」。
<code>vsn</code>	指定給磁碟區的磁碟區序列名稱 (VSN)。

2. 輸入 `auditslot(1M)` 指令，以更新剩餘空間資訊。

```
auditslot -e eq:slot[:partition]
```

表 2-7 `auditslot(1M)` 的引數

引數	意義
<code>-e</code>	如果指定 <code>-e</code> 選項並且媒體是磁帶，便會更新剩餘空間。否則，其不會變更。
<code>eq</code>	如 <code>mcf</code> 檔案所定義之自動程式庫或手動載入之磁碟機的設備序號。
<code>slot</code>	自動程式庫中，程式庫目錄中所指出的儲存插槽號碼。此引數不適用於手動載入磁碟機。
<code>partition</code>	磁光碟的一面。分割區必須是 1 或 2。此引數不適用於磁帶卡匣。

如需有關 `auditslot(1M)` 指令的更多資訊，請參閱「`auditslot(1M)` 線上手冊」。您也可以使用 `samu(1M)` 公用程式的 `:audit` 指令或 `File System Manager` 執行此作業。

▼ 從磁碟機中移除卡住的卡匣

如果卡匣被卡在磁碟機中，請遵循以下步驟。

1. 使用 `samcmd(1M) off` 指令關閉自動程式庫中的磁碟機。

```
samcmd off eq
```

對於 `eq`，請依據 `mcf` 檔案中所定義，指定磁碟機的設備序號。
您也可以使用 `samu(1M)` 或 File System Manager 執行此步驟。

2. 使用 `samcmd(1M) off` 指令關閉自動程式庫。

```
samcmd off eq
```

對於 `eq`，請依據 `mcf` 檔案中所定義，指定程式庫的設備序號。
您也可以使用 `samu(1M)` 或 File System Manager 執行此步驟。

3. 從磁碟機中實際移除卡匣。
確定卡匣或磁碟機未損毀。

4. 使用 `samcmd(1M) on` 指令開啟自動程式庫以及磁碟機。
針對磁碟機以及程式庫各執行一次此指令。

```
samcmd on eq
```

對於 `eq`，請依據 `mcf` 檔案中所定義，指定程式庫或磁碟機的設備序號。
如果自動程式庫在開啓時執行稽核，則您已完成所有步驟。

5. 自動程式庫若未在開啟時執行稽核，請執行下列步驟：

a. 將卡匣放回其儲存插槽。

b. 使用 `chmed(1M)` 指令調整程式庫目錄，以設定損毀磁帶的佔用旗標。

請用以下格式使用此指令：

```
chmed +o eq:slot
```

表 2-8 `chmed(1M)` 的引數

引數	意義
<code>eq</code>	如 <code>mcf</code> 檔案所定義之自動程式庫或磁碟機的設備序號。
<code>slot</code>	程式庫中，程式庫目錄所指出的儲存插槽號碼。此引數不適用於手動載入磁碟機。

如需有關 `chmed(1M)` 指令的更多資訊，請參閱「`chmed(1M)` 線上手冊」。

如果將卡匣留在外面，並想稍後將其裝回，則必須將卡匣匯入自動程式庫。

目錄檔作業，匯入和匯出卡匣

實際向自動程式庫增加 (匯入) 或從中移除 (匯出) 卡匣可以讓您執行包括下列功能在內的數項功能：

- 更換卡匣。
- 將卡匣易地存放，以便日後進行災難回復時使用。您可以對 `chmed(1M)` 指令使用 `-I` 選項，以指定諸如卡匣儲存位置之類的額外資訊。

當匯入和匯出卡匣時，您也可以更新程式庫目錄檔。

程式庫目錄是集中儲存庫，其包含了 Sun StorageTek SAM 環境在自動程式庫中尋找卡匣時所必需的所有資訊。程式庫目錄檔是二進位 UNIX 檔案系統 (UFS) 常駐檔案，其中包含有關自動程式庫中每個插槽的資訊。此檔案中的資訊包括：一個或多個與插槽中儲存的卡匣相關聯的 `VSN`、該卡匣中的剩餘容量與空間、用於指示唯讀、防寫保護、回收的旗標，以及有關卡匣的其他狀態資訊。

Sun StorageTek SAM 環境會根據自動程式庫連接伺服器的方式，以如下所示的不同方法處理目錄：

- 若為直接連結式自動程式庫，則程式庫目錄會在程式庫目錄項目與自動程式庫的實體插槽進行一對一的對映。程式庫目錄中的第一條項目對應於自動程式庫中的第一個插槽。在需要卡匣時，系統將查閱程式庫目錄檔以確定包含 VSN 的插槽，並輸入指令以將卡匣從該插槽載入磁碟機。

備註 – Sun StorageTek SAM 在直接連結式程式庫中不支援使用混用的媒體。如果程式庫經過分割，則每個分割區必須僅包含一種媒體類型。

- 若是網路連結式自動程式庫，程式庫目錄即不會直接與插槽對映。而是已知安裝在自動程式庫中的 VSN 之清單。在需要卡匣時，系統會傳送請求至供應商軟體以將 VSN 下載至磁碟機。供應商軟體將找出 VSN 儲存插槽所在位置。

由於系統特性與供應商提供的軟體不同，因此每個自動程式庫處理卡匣匯入與匯出的方式也不同。例如，在 ACL 4/52 程式庫中，您需要在從自動程式庫中匯出卡匣之前，執行 `move` 指令以便將卡匣移至匯入/匯出單元中。

網路連結式自動程式庫可使用自有的公用程式匯入與匯出卡匣，因此 `import(1M)` 和 `samexport(1M)` 指令只會更新 Sun StorageTek SAM 系統所使用的程式庫目錄項目。您若是使用網路連結式程式庫，可參閱第 205 頁的「[使用供應商指定作業程序的程式庫基本作業](#)」，以取得有關匯入與匯出卡匣的資訊。

追蹤匯出的媒體 — 記錄器

Sun StorageTek SAM 記錄器會持續追蹤從自動程式庫或手動掛載之裝置上匯出的卡匣。記錄器的功能與虛擬程式庫相同，但它沒有定義的硬體裝置。與自動程式庫相同，它也是在 `mcf` 檔案中配置、具有記錄與其相關的所有卡匣項目的目錄、可以匯入與匯出卡匣，並可以在 File System Manager 中顯示為另一個自動程式庫。

記錄器可以使用 `hy` 裝置類型在 `mcf` 檔案中配置。如果並未在 `mcf` 檔案中配置記錄器，便會使用如下方式建立：

```
historian n+1 hy - on /var/opt/SUNWsamfs/catalog/historian
```

在之前的項目中，`n+1` 是 `mcf` 檔案中最後的一個設備序號加 1。若要對目錄使用其他設備序號或路徑名稱，需要在 `mcf` 中定義記錄器。

記錄器程式庫目錄檔在記錄器第一次啟動時初始化並具有 32 個項目。確定目錄檔位於足夠大的檔案系統（可以容納整個目錄檔）中。您的站點可能想要追蹤程式庫匯出的現有 Sun StorageTek SAM 卡匣。在此情況下，您需要按照「[build_cat\(1M\)](#) 線上手冊」中所述，從現有卡匣建立記錄器目錄。

defaults.conf 檔案中的以下兩種配置指令可以影響記錄器的運作方式：

- 若是使用 `exported_media = unavailable` 指令，則所有從自動程式庫匯出的卡匣，均會標記為無法供記錄器使用。標記為無法使用的卡匣之要求將產生 EIO 錯誤。
- 如果出現 `attended = no` 指令，它會向記錄器宣告說沒有處理載入請求的操作者。記錄器已知的載入卡匣與尚未載入的卡匣之請求將產生 EIO 錯誤。

如需配置詳細資訊，請參閱「[historian\(7\) 線上手冊](#)」和「[defaults.conf\(4\) 線上手冊](#)」。

從自動程式庫匯入與匯出

信箱是自動程式庫中用以增加及移除卡匣的區域。`import(1M)` 指令可將卡匣從信箱移至儲存插槽。`samexport(1M)` 指令可將卡匣從儲存插槽移至信箱。對大多數程式庫而言，若卡匣在 Sun StorageTek SAM 軟體啟動時即位於信箱內，則軟體即會在啟動時自動匯入卡匣。

匯入與匯出實際操作會因製造商而異。如需判別您的設備是否需執行特定的程序，請參閱第 205 頁的「[使用供應商指定作業程序的程式庫基本作業](#)」。

以下章節說明匯入與匯出卡匣：

- [第 26 頁的「從使用信箱的程式庫匯入卡匣」](#)
- [第 27 頁的「從使用信箱的程式庫匯出卡匣」](#)
- [第 27 頁的「自不使用信箱的程式庫匯入卡匣」](#)
- [第 28 頁的「自不使用信箱的程式庫匯出卡匣」](#)

▼ 從使用信箱的程式庫匯入卡匣

1. 使用製造商建議的作業開啟信箱。

信箱旁邊通常會有一個按鈕。信箱有時是單插槽信箱，在供應商的文件資料中被稱為郵件插槽。

2. 手動將卡匣放入信箱。
3. 關閉信箱。
4. 使用 `import(1M)` 指令匯入卡匣。

```
import eq
```

對於 `eq`，請依據 `mcf` 檔案中所定義，指定程式庫的設備序號。系統會將卡匣從信箱移至儲存插槽，並更新卡匣的程式庫目錄。您也可以使用 `samu(1M)` 或 File System Manager 執行此作業。

▼ 從使用信箱的程式庫匯出卡匣

1. 使用 `samexport(1M)` 指令將卡匣從儲存插槽移至信箱。

以下列格式之一使用此指令：

```
samexport eq:slot  
samexport media-type.vsn
```

表 2-9 samexport(1M) 的引數

引數	意義
<i>eq</i>	如 <code>mcf</code> 檔案所定義之自動程式庫的設備序號。
<i>slot</i>	自動程式庫中，程式庫目錄中所指出的儲存插槽號碼。
<i>media-type</i>	卡匣的媒體類型。如需有效媒體類型清單，請參閱「 <code>mcf(4)</code> 線上手冊」。
<i>vsn</i>	指定給磁碟區的磁碟區序列名稱 (VSN)。

您也可以使用 `samu(1M)` 或 File System Manager 執行此步驟。

2. 使用製造商建議的作業來開啟信箱或郵件插槽。

信箱旁邊通常會有一個按鈕。

▼ 自不使用信箱的程式庫匯入卡匣

1. 使用 `samcmd(1M) unload` 指令。

```
samcmd unload eq
```

對於 *eq*，請依據 `mcf` 檔案中所定義，指定程式庫的設備序號。

2. 待系統完成目前的作業之後，會將其狀態設定為 `off`，再將目前使用中的目錄傳輸至記錄器。
3. 解除鎖定並打開自動程式庫的門蓋。
4. 將卡匣載入可用插槽。
5. 合上並鎖定自動程式庫的門蓋。

自動程式庫將重新初始化並掃描程式庫中的卡匣。Sun StorageTek SAM 軟體會更新程式庫目錄檔，將匯入卡匣的 VSN 新增至目錄。自動程式庫的狀態將設定為 `on`。

▼ 自不使用信箱的程式庫匯出卡匣

1. 使用 `samcmd(1M)` `unload` 指令。

```
samcmd unload eq
```

對於 `eq`，請依據 `mcf` 檔案所定義，指定要定址之程式庫的設備序號。

2. 待系統完成目前的作業之後，會將其狀態設定為 `off`，再將目前使用中的目錄傳輸至記錄器。
3. 解除鎖定並打開自動程式庫的門蓋。
4. 將卡匣從各自的插槽中移除。
5. 合上並鎖定自動程式庫的門蓋。

自動程式庫將重新初始化並掃描自動程式庫中的卡匣。系統將使用目前程式庫插槽中卡匣的 `VSN` 來更新程式庫目錄檔。被移除卡匣的 `VSN` 已從程式庫目錄檔中移除，且現在只記錄於記錄器檔案中。自動程式庫的狀態將設定為 `on`。

▼ 啓動載入通知

Sun StorageTek SAM 軟體會要求定期載入卡匣，以滿足歸檔與分段備份需求。如果請求載入位於程式庫內部的卡匣，則系統會自動處理請求。如果請求載入位於程式庫外部的卡匣，則需要操作者進行操作。如果已啓動載入通知，則當需要從程式庫外取得卡匣時，`load_notify.sh` 程序檔會傳送電子郵件。

1. 成為超級使用者。
2. 使用 `cp(1)` 指令將載入通知程序檔從其安裝位置複製到可操作位置。

例如：

```
# cp /opt/SUNWsamfs/examples/load_notify.sh  
/etc/opt/SUNWsamfs/scripts/load_notify.sh
```

3. 使用 `more(1)` 或另一個指令來檢查 `defaults.conf` 檔案。

請確定檔案中存在以下指令並且如下所示：

- `exported_media=available`
- `attended=yes`

這些指令是依據預設而設定。如果載入通知功能已啟動，請確定這些指令未作任何變更。

4. 修改 `load_notify.sh` 程序檔以便將通知傳送給操作者。

依據預設，程序檔會將電子郵件傳送至 `root`，但是也可編輯它以將電子郵件傳送至其他人員，或撥打呼叫器，或提供某些其他通知方式。

使用具備加密功能的磁碟機

如果您要將檔案歸檔至具備加密功能磁碟機，以下是幾點特殊的注意事項：

- 您不應在程式庫中混用無加密功能與具備加密功能的磁碟機。
- 磁碟機一旦啓用了加密功能，便無法停用。
- 請勿在磁帶上混用加密與未加密的檔案。
- 加密的磁碟機無法附加至包含未加密資料的磁帶。
- 但是，啓用加密功能的磁碟機可以讀取未加密的資料。

因為具有這些限制，所以如果您在歸檔系統中同時使用了未加密與加密的資料及磁碟機，則必須小心規劃 `archiver.cmd` 檔案中的歸檔集和磁帶指定。

手動載入磁碟機作業

本節說明這些作業會在您具有手動載入獨立式磁碟機 (而非自動程式庫) 時有所不同。每個手動載入的磁碟機均有自己的單插槽程式庫目錄檔。

▼ 載入卡匣

- 若要將卡匣載入手動載入裝置，請根據製造商的指示將卡匣放入磁碟機。

Sun StorageTek SAM 系統會確定卡匣載入與否、讀取標籤，並更新單一插槽的目錄。無需採取其他動作。

▼ 卸載卡匣

- 使用 `samcmd(1M) idle` 指令以閒置磁碟機。

```
samcmd idle eq
```

對於 `eq`，請依據 `mcf` 檔案中所定義，指定磁碟機的設備序號。

本指令可確保沒有已啟動的歸檔或分段備份程序。在所有 I/O 作業完成時，磁碟機將從 `idle` 切換至 `off`，然後退出磁帶。

卡匣若為磁帶，即會進行倒帶，並可隨時移除。光碟會自動退出。請參閱製造商對有關移除特定卡匣的指示。

您也可以使用 `samu(1M)` 或 File System Manager 執行此作業。

▼ 檢視程式庫目錄檔

- 使用 `samu(1M)` 公用程式的 `:v` 指令。

```
:v eq
```

對於 `eq`，請依據 `mcf` 檔案中所定義，指定程式庫的設備序號。

第3章

歸檔

歸檔是將檔案從 Sun StorageTek SAM 檔案系統複製到位於可移除式媒體匣或其他系統磁碟分割區中之磁碟區的過程。使用 Sun StorageTek SAM 歸檔功能可讓您指定要立即歸檔或永不歸檔的檔案，以及執行其他作業。

在本章中，**歸檔媒體**這個專有名詞指的是歸檔磁碟區寫入的各種卡匣或磁碟區塊。本章將說明歸檔器的作業理論，提供對開發站台歸檔規則的一般指引，及解釋如何透過建立 `archiver.cmd` 檔案以執行規則。

本章包含以下各節：

- 第 32 頁的「歸檔程序簡介」
- 第 42 頁的「關於 `archiver.cmd` 檔案」
- 第 45 頁的「使用歸檔器指令」
- 第 82 頁的「關於磁碟歸檔」
- 第 90 頁的「計畫歸檔作業」
- 第 91 頁的「歸檔器範例」

歸檔程序簡介

歸檔器會自動將 Sun StorageTek SAM 檔案寫入歸檔媒體。無需操作者介入，即可進行檔案歸檔。檔案會歸檔到歸檔媒體上的磁碟區中，而每一個磁碟區則會由名為磁碟區序列名稱 (VSN) 的唯一識別碼所識別。歸檔媒體可包含一個或多個磁碟區。

在掛載 Sun StorageTek SAM 檔案系統時，歸檔器就會自動開始執行。您可將歸檔指令加入以下檔案，以自訂站點的歸檔器作業。

```
/etc/opt/SUNWsamfs/archiver.cmd
```

archiver.cmd 檔案無需為要進行的歸檔保留。在沒有這個檔案的時候，歸檔器會採用以下預設：

- 所有檔案將歸檔至可用的磁碟區。
- 所有檔案的歸檔時間為 4 分鐘。歸檔時間是自檔案上次修改後的時間。
- 歸檔間隔為 10 分鐘。歸檔間隔是完成歸檔處理所需的時間。

下列章節將說明歸檔組概念，並解釋在歸檔處理時執行的操作。

歸檔器監控程序

sam-archiverd 常駐程式會排定歸檔作業。sam-arfind 程序會將要歸檔的檔案指定給歸檔組。sam-arcopy 程序會將要歸檔的檔案複製到選定磁碟區。

sam-archiverd 常駐程式是在 Sun StorageTek SAM 作業開始時，由 sam-fsd 所啟動的。sam-archiver 常駐程式將執行 archiver(1M) 指令以讀取 archiver.cmd 檔案，並建立控制歸檔所需的表格。其會針對所掛載的各個檔案系統啟動 sam-arfind 程序；如有檔案系統卸載，便會停止相關的 sam-arfind 程序。sam-archiverd 程序將監控 sam-arfind 與來自操作者或其他程序的程序訊號。

歸檔組

歸檔組是指要進行歸檔的一組檔案。歸檔組可透過任何檔案系統群組定義。歸檔組中的檔案共享與大小、所有權、群組或目錄位置有關的一般條件。歸檔組可控制歸檔備份的目標、歸檔備份的保存時間，以及歸檔資料之前軟體的等待時間。歸檔組中的所有檔案將複製到與此歸檔組相關的磁碟區。檔案系統中的檔案只可隸屬於一個歸檔組。

在建立與修改檔案時，歸檔器會將它們複製到歸檔媒體。歸檔程序也會複製 Sun StorageTek SAM 檔案系統作業所需的資料，包括目錄、符號連結、區段檔案的索引，以及歸檔媒體資訊。

歸檔檔案與標準的 UNIX tar(1) 格式相容。如此可確保資料與 Solaris 作業系統和其他 UNIX 系統相容。如果您的 Sun StorageTek SAM 環境發生完全消失的狀況，tar(1) 格式會允許使用標準 UNIX 工具和指令來修復檔案。

歸檔組名稱由管理員決定，且除了下列情況之外，幾乎不受任何限制：

- 具有兩個預留的歸檔組名稱：`no_archive` 和 `allsets`。
依據預設，系統將定義 `no_archive` 歸檔組。在此歸檔組中選擇的檔案永不歸檔。例如，暫存目錄 (如 `/sam1/tmp`) 中的檔案可能包含在 `no_archive` 歸檔組中。
`allsets` 歸檔組可用於定義要套用到所有歸檔組上的參數。
- 為每個 Sun StorageTek SAM 檔案系統所命名的歸檔組會預留用於控制結構資訊。Sun StorageTek SAM 檔案系統會為每個檔案系統提供預設歸檔組。對於每個檔案系統，中介資料與資料檔將歸檔。檔案系統歸檔組將包含目錄與連結資訊及其他歸檔組中不包含的任何檔案。系統將給予預設的歸檔組相關檔案系統的名稱，且無法變更。例如，對於配置與命名為 `samfs1` 的檔案系統，`samfs1` 即為歸檔組名稱。
- 歸檔組名稱限制為 29 個字元。字元限制為 26 個大寫與小寫字母、數字 0 到 9 及底線字元 (`_`)。

歸檔操作

歸檔器預設會為每一個歸檔組製作一個備份，但您最多只可請求四個備份。歸檔組與備份數目將成為磁碟區集合的同義字。歸檔備份提供對獨立磁碟區上的檔案進行複製。

在將檔案視為歸檔或重新歸檔的候選者之前，檔案中的資料必須修改。僅存取檔案不會將其歸檔。例如，在檔案上執行 `touch(1)` 或 `mv(1)` 指令不會導致檔案歸檔或重新歸檔。

備註 — 執行 `mv(1)` 指令將變更檔案名稱，但不會變更檔案資料；同時如果從 `tar(1)` 檔案中回復，這將具有災難復原狀況的分支。如需有關災難復原的更多資訊，請參閱「Sun StorageTek Storage Archive Manager 疑難排解指南」。

選取檔案進行歸檔時，會以檔案的歸檔時間為準，亦即檔案自上次修改後所經過的時間。歸檔時間可為每個歸檔備份進行定義。

使用者可使用 `touch(1)` 指令將其檔案的預設時間參照變更為過去或將來的值。但是，這樣會導致不預期的歸檔結果。為避免發生此問題，歸檔器會調整參照，使其一律介於檔案建立時間與目前的時間之間。

歸檔優先權可從檔案屬性特性，及從與歸檔組相關的檔案屬性倍增器計算。基本上，計算如下：

$$\text{archive-priority} = \text{file-property-value} \times \text{property-multiplier}$$

file-property-value 的數字大多為 1 (代表 `true`) 或 0 (代表 `false`)。例如，如果製作了歸檔備份 1，則屬性備份 1 的值是 1。備份 2、備份 3 與備份 4 的值則為 0。至於歸檔時間與檔案大小等其他特性的值，則可以是 0 或 1 以外的值。

property-multiplier 值取決於歸檔組的 `-priority` 參數。您可以針對檔案的各個方面 (如時間或大小) 指定值，以決定歸檔請求的優先權。如需有關 `-priority` 參數的更多資訊，請參閱「`archiver.cmd(4)` 線上手冊」。

archive-priority 與 *property-multiplier* 值均為浮點數字。所有屬性倍增器的預設值是 0.0。歸檔請求在歸檔請求中被設定為最高檔案優先權。

下列章節說明歸檔器從初始的檔案掃描到檔案備份處理所執行的步驟。

步驟 1：辨識要歸檔的檔案

每個掛載的檔案系統都有單獨的 `sam-arfind` 程序。`sam-arfind` 程序會監視每個檔案系統，以決定需要歸檔的檔案。每當檔案受到變更且此變更會影響其歸檔狀態時，檔案系統就會通知其 `sam-arfind` 程序。這種變更的範例包括檔案修改、重新歸檔、取消歸檔及重新命名。收到通知時，`sam-arfind` 程序會檢查檔案以決定所需執行的歸檔動作。

`sam-arfind` 程序可使用檔案屬性說明以決定檔案所屬的歸檔組。用於決定檔案的歸檔組之特性包含如下：

- 使用常規表示式的檔案名稱的目錄路徑部分及完整檔案名稱 (選用)
- 檔案所有者的使用者名稱
- 檔案所有者的群組名稱
- 最小檔案大小
- 最大檔案大小

當達到或超出一或多個備份的檔案歸檔時間時，`sam-arfind` 即會將檔案增加到歸檔組的一或多個歸檔請求中。歸檔請求是所有屬於相同歸檔組之檔案的集合。歸檔請求位於下列目錄：

```
/var/opt/SUNWsamfs/archiver/file_sys/ArchReq
```

此目錄中的檔案為二進位檔案，而且您可以藉由使用 `showqueue(1M)` 指令來顯示這些檔案。

個別的歸檔請求會用於尚未歸檔及需要重新歸檔的檔案上。如此即可個別控制這兩種類型之檔案的排程作業。

如果還沒達到一或多個備份的檔案歸檔時間，檔案位於的目錄和達到歸檔時間的時間會新增到掃描清單中。當到達掃描清單的時間，就會對目錄進行掃描。到達其歸檔時間的檔案會新增到歸檔請求中。

如果檔案已離線，`sam-arfind` 程序將為歸檔備份選擇用作來源的磁碟區。如果檔案備份正在重新歸檔，`sam-arfind` 程序將選擇包含正在重新歸檔的歸檔備份之磁碟區。

如果檔案已分段，則僅選擇歸檔那些已變更的區段。區段檔案的索引不包含使用者資料，因此它將被視為檔案系統歸檔組的成員並單獨歸檔。

可使用以下兩種方法標示要進行歸檔的檔案：**連續歸檔**和**掃描歸檔**。若是連續歸檔，歸檔器會和檔案系統一起運作，決定哪些檔案需要歸檔。若是掃描歸檔，歸檔器會定期詳察檔案系統並選取要歸檔的檔案。以下章節將說明這兩種方法。

連續歸檔

連續歸檔為預設的歸檔方法 (`archiver.cmd` 檔案參數為 `examine=noscan`)。藉由連續歸檔，您可以使用 `-startage`、`-startcount` 與 `-startsize` 參數指定歸檔組排程的開始條件。這些條件可讓您找出最佳的時機進行歸檔工作。例如：

- 若建立應歸檔在一起的檔案需費時 1 小時，即可將 `-startage` 參數設為 1 小時 (`-startage 1h`)，以確保會在排程歸檔請求之前建立所有檔案。
- 您可以將 `-startsize` 指定為 150 GB (`-startsize 150g`)，以指示歸檔器等候資料達到 150 GB 的大小時，才開始進行歸檔。
- 若已知將會產生 3000 個檔案進行歸檔，即可指定 `-startcount 3000`，以確保檔案會歸檔在一起。

只要達到任一項排程開始條件，`sam-arfind` 程序便會將每一個歸檔請求傳送至歸檔器常駐程式 (`sam-archiverd`) 進行排程，以將檔案複製到歸檔媒體。

如需有關歸檔參數的更多資訊，請參閱第 46 頁的「[全域歸檔指令](#)」。

備註 – 當 `examine` 設定為 `noscan` 時，便會自動實作下列預設設定：

- `startage` : 10 分鐘
- `startsize` : 10 GB
- `startcount` : 10,000 個檔案

掃描歸檔

此為連續歸檔的替代方法；您可以在 `archiver.cmd` 檔案中指定 `examine=scan`，以指示 `sam-arfind` 藉掃描的方式檢查所要歸檔的檔案。需要歸檔的檔案會置入歸檔請求。`sam-arfind` 程序將定期掃描每個檔案系統，以決定需要歸檔的檔案。第一次會掃描目錄，此時 `sam-arfind` 將以遞迴方式向下掃描樹狀目錄。此程序會檢查每個檔案，並將不需要歸檔之檔案的檔案狀態旗標設為「`archdone is set`」。在連續掃描期間，`sam-arfind` 會掃描 `.inode` 檔案。只有未設定 `archdone` 旗標的 `inode` 會施以檢查。

如需有關控制 `archdone` 旗標設定的資訊，請參閱第 54 頁的「[setarchdone 指令：控制 archdone 旗標的設定](#)」。

完成檔案系統掃描時，`sam-arfind` 程序會將每個歸檔請求傳送至歸檔器常駐程式 (`sam-archiverd`) 進行排程，以將檔案複製到歸檔媒體。接著，在 `interval=time` 指令所指定的持續時間內，`sam-arfind` 程序將會靜止。當間隔結束時，`sam-arfind` 程序將繼續掃描。

步驟 2：組合歸檔請求

當 `sam-archiverd` 常駐程式收到歸檔請求時，會將這些歸檔請求加以組合。本節說明組合程序。

歸檔請求中的檔案可能會因為歸檔媒體的容量不足，或歸檔器指令檔中所指定的控制不足，而無法同時進行歸檔。組合是指從歸檔請求中選取檔案一併歸檔的過程。當歸檔請求的歸檔備份作業完成時，若檔案仍要歸檔，便會重新組合歸檔請求。

`sam-archiverd` 常駐程式會依據特定的預設值與站點特定條件，將檔案置入歸檔請求中。預設作業是依照掃描檔案系統時檔案的出現順序，將歸檔請求中的所有檔案歸檔至相同的歸檔磁碟區。站點特定條件可讓您控制檔案的歸檔順序，以及檔案在磁碟區上的分佈方式。這些條件稱為歸檔組參數，會以下列順序進行評估：`-reserve`、`-join`、`-sort`、`-rsort` (反向排序) 與 `-drives`。如需有關這些參數的更多資訊，請參閱「`archiver.cmd(4)` 線上手冊」。

如果歸檔請求屬於指定 `-reserve owner` 的歸檔組，`sam-archiverd` 常駐程式將根據檔案目錄路徑、使用者名稱或群組名稱對歸檔請求中的檔案進行排序。將選擇屬於第一個 `owner` 的檔案以歸檔。剩餘的檔案以後歸檔。

歸檔請求所屬的歸檔組若有指定 `-join method`，則 `sam-archiverd` 常駐程式便會依據指定的合併方法組合檔案。若同時指定了 `-sort` 或 `-rsort method`，`sam-archiverd` 常駐程式便會依據指定的排序方法排序各群組中的檔案。在接下來的組合與排定程序中，會將合併檔案中的各群組視為單一檔案。

歸檔請求所屬的歸檔組若有指定 `-sort` 或 `-rsort method`，則 `sam-archiverd` 常駐程式便會依據指定的排序方法排序檔案。視排序方法而定，`sam-archiverd` 常駐程式將基於排序方法、時間、大小或目錄位置傾向於將檔案保留在一起。歸檔請求預設不會排序，因此檔案將會依照檔案系統掃描時出現的順序進行歸檔。

`sam-archiverd` 常駐程式決定檔案在線上或離線。如果線上檔案與離線檔案均在歸檔請求中，將首先選擇線上檔案以歸檔。

歸檔請求若無需依照排序方法進行合併或排序，則離線檔案便會由歸檔備份所在的磁碟區進行排序。如此可確保相同磁碟區上之各歸檔組中的所有檔案，會同時以其儲存到媒體上的順序進行分段備份。在製作離線檔案的多個歸檔備份時，離線檔案將直到製作所有需要的備份時才釋放。與第一個檔案處於相同磁碟區的所有待分段備份檔案均將被選定用於歸檔。

備註 – 歸檔的檔案順序若與離線檔案所需的磁碟區順序不符，則 `-join`、`-sort` 與 `-rsort` 參數即可能對離線檔案歸檔作業的效能產生負面的影響。僅在第一次進行歸檔備份時才使用這些參數。開始備份時若有足夠的歸檔媒體空間可用，則其他備份便會繼續使用第一個備份的順序。

歸檔請求在組合之後，便會進入 `sam-archiverd` 常駐程式的排程佇列中，如下節所說明。

步驟 3：排定歸檔請求

當發生下列一種情況時，`sam-archiverd` 常駐程式中的排程程式會依據需要而執行：

- 歸檔請求將輸入排定佇列中。
- 已完成對歸檔請求的歸檔。
- 將從目錄伺服器接收媒體狀態的變更。
- 將接收到歸檔器狀態變更的訊息。

排定佇列中的歸檔請求將按優先權進行排序。排程程式會在每次執行時檢查所有的歸檔請求，以決定是否可將這些請求指定給 `sam-arcopy` 程序，進而將檔案複製到歸檔媒體。

下列條件必須成立，才可排定歸檔請求：

- 必須有磁碟機可用於進行檔案備份。
- 必須有歸檔組可用的磁碟區，且其具有足夠的空間，可用於儲存歸檔請求中的檔案。

磁碟機

如果歸檔組已指定 `-drives` 參數，`sam-archiverd` 常駐程式將切分多個磁碟機中的歸檔請求之選定檔案。此時可用的磁碟機數目若少於 `-drives` 參數指定的數目，便會使用較小的數目。

如果歸檔請求的檔案總量小於 `-drivemin` 值，將僅使用一個磁碟機。`-drivemin` 值是 `drivemin` 參數指定的值或 `archmax` 值。`archmax` 值是由 `-archmax` 參數所指定，或為媒體定義的值。如需有關 `-archmax` 參數與 `archmax=` 指令的更多資訊，請參閱「`archiver.cmd(4)` 線上手冊」。

歸檔請求中的檔案大小總和若是超出 `-drivemin` 值，則所使用的磁碟機數量，便會是該檔案大小總和除以 `-drivemin` 值所得的值。所使用的磁碟機數量若是小於 `-drives` 參數所指定的磁碟機數量，便會使用前者的值。

磁碟機進行檔案歸檔所需的時間不盡相同。您可以利用 `-drivemax` 參數提升磁碟機的使用效能。`-drivemax` 參數會要求您先指定要寫入磁碟機的最大位元組數，才可為該磁碟機重新排定更多資料。

磁碟區

若要執行歸檔，至少需具備一個空間足以容納歸檔請求中之部分檔案的磁碟區。其空間若是足夠，便會排定最近用於歸檔組的磁碟區。此時歸檔器必須不在使用該磁碟區。

歸檔組可使用的磁碟區若已在使用中，除非您指定了 `-fillvsns` 參數，否則便會選取其他磁碟區。在此情況下，無法排程歸檔請求。

如果歸檔請求對於一個磁碟區太大，則將選擇符合此磁碟區的檔案以將其歸檔至磁碟區。如果歸檔請求包含大於一個磁碟區的檔案，且沒有選擇歸檔請求的磁碟區溢位功能，則檔案將無法歸檔。適合此情況的訊息將傳送至記錄檔。

您可以分別使用 `-ovflmin` 參數或 `ovflmin=` 指令指定歸檔組或媒體的磁碟區溢位。如需有關 `-ovflmin` 參數與 `ovflmin=` 指令的更多資訊，請參閱「archiver.cmd(4) 線上手冊」。 `ovflmin` 指定可決定檔案大小的臨界值；當高於該值時，便會為歸檔作業指定額外的磁碟區或媒體。為歸檔組所指定的 `ovflmin` 值，優先於為媒體指定的 `ovflmin` 值。

檔案大小若小於 `ovflmin` 的值，便無法進行歸檔。適合此情況的訊息將傳送至記錄檔。檔案大小若大於 `ovflmin` 的值，便會依據需要指定額外的磁碟區。選取磁碟區時會先挑選容量大者，以儘可能減少所需的磁碟區數量。如果找不到歸檔請求可使用的磁碟區，歸檔請求會等待，直至磁碟區變為可用為止。

在決定特定歸檔請求的排定優先權時，某些特性 (如檔案在線上或離線) 可與歸檔優先權搭配使用。如需有關自訂優先權倍增器的更多資訊，請參閱「archiver.cmd(4) 線上手冊」中的 `-priority` 參數。

對於每個歸檔請求，`sam-archiverd` 常駐程式會將歸檔優先權新增至與各種系統資源屬性相關的倍增器以計算排定優先權。這些特性與佇列歸檔請求的秒數、要在歸檔程序中使用的第一個磁碟區是否載入磁碟機等有關。

`sam-archiverd` 常駐程式可使用調整的優先權指定每一個已可進行複製的歸檔請求，如下節所說明。

步驟 4：為歸檔請求中的檔案進行歸檔

當歸檔請求已可進行歸檔時，`sam-archiverd` 常駐程式會標記歸檔檔案 (`tarball`) 範圍，使每個歸檔檔案的大小小於指定的 `-archmax` 值。如有單一檔案大於此值，其便會成為歸檔檔案中唯一的檔案。

對於每個歸檔請求與每個要使用的磁帶機，`sam-archiverd` 常駐程式會將歸檔請求指定給 `sam-arcopy` 程序，以便將檔案複製到歸檔媒體。歸檔資訊將輸入 `inode`。

如果已啟用歸檔記錄，將建立歸檔記錄。

每個檔案在分段備份之後，便會釋放其所佔用的磁碟空間，直到清單中的所有檔案皆完成歸檔為止。

各種錯誤與檔案狀態變更會阻止檔案成功複製，這些錯誤可能包括讀取快取磁碟時所發生的錯誤，以及寫入磁碟區時所發生的錯誤。狀態變更可能包括選取之後有所修改、檔案開啓供人寫入，或檔案遭到移除。

在 `sam-arcopy` 程序結束後，`sam-archiverd` 常駐程式將檢查歸檔請求。如果有任何檔案尚未歸檔，歸檔請求將重新組合。

您也可以 `archiver.cmd` 檔案中指定歸檔作業的逾時。指令如下所示：

```
timeout = [operation | media] time
```

operation 可能是下列其中之一：

- *read*：從磁碟讀取檔案。預設逾時為 1 分鐘。
- *request*：請求歸檔媒體。預設逾時為 15 分鐘。
- *stage*：分段備份要歸檔的檔案。預設逾時為 15 分鐘。
- *write*：寫入歸檔媒體。預設逾時為 15 分鐘。

亦可為個別媒體指定寫入作業的逾時。

歸檔請求佇列時間通知

您可以配置 `archiver.cmd` 檔案中的 `-queue_time_limit time` 歸檔組參數，以在排程佇列中的歸檔請求超過一定時間長度後，通知管理員。如果歸檔請求在指定時間結束後仍在佇列中，則會傳送一封電子郵件給系統管理員。

範例預設執行結果

程式碼範例 3-1 會顯示 `archiver(1M) -l` 指令的輸出範例。

程式碼範例 3-1 `archiver(1M) -l` 指令的輸出

```
# archiver
Archive media:
default:mo
media:mo archmax:5000000
media:lt archmax:50000000
Archive devices:
device:mo20 drives_available:1 archive_drives:1
device:lt30 drives_available:1 archive_drives:1
Archive file selections:
Filesystem samfs1:
samfs1 Metadata
    copy:1 arch_age:240
big path:. minsize:512000
    copy:1 arch_age:240

all path:
    copy:1 arch_age:30
Archive sets:
all
    copy:1 media:mo
big
    copy:1 media:lt
samfs1
    copy:1 media:mo
```

歸檔記錄檔案與事件記錄

sam-arfind 與 sam-arcopy 程序可使用 syslog 功能與 archiver.sh，將警告與參考訊息記錄在內含已歸檔或自動取消歸檔之各檔案相關資訊的記錄檔中。記錄檔是歸檔動作的連續記錄。您可使用記錄檔以找出作為傳統備份用途的較早檔案備份。

根據預設值，將不產生此檔案。您可以在 archiver.cmd 檔案中使用 logfile= 指令指定所要建立的記錄檔及記錄檔的名稱。如需有關記錄檔的更多資訊，請參閱本章第 45 頁的「使用歸檔器指令」，並參閱「archiver.cmd(4) 線上手冊」。

程式碼範例 3-2 顯示定義每個欄位的歸檔器記錄檔之文字行範例。

程式碼範例 3-2 歸檔器記錄檔文字行

```
A 2001/03/23 18:42:06 mo 0004A arset0.1 9a089.1329 samfs1 118.51
162514 t0/fdn f 0 56
A 2001/03/23 18:42:10 mo 0004A arset0.1 9aac2.1 samfs1 189.53
1515016 t0/fae f 0 56
A 2001/03/23 18:42:10 mo 0004A arset0.1 9aac2.b92 samfs1 125.53
867101 t0/fai f 0 56
A 2001/03/23 19:13:09 lt SLOT22 arset0.2 798.1 samfs1 71531.14
1841087 t0/fhh f 0 51
A 2001/03/23 19:13:10 lt SLOT22 arset0.2 798.e0e samfs1 71532.12
543390 t0/fhg f 0 51
A 2003/10/23 13:30:24 dk DISK01/d8/d16/f216 arset4.1 810d8.1 qfs2
119571.301 1136048 t1/fileem f 0 0
A 2003/10/23 13:30:25 dk DISK01/d8/d16/f216 arset4.1 810d8.8ad
qfs2 119573.295 1849474 t1/fileud f 0 0
A 2003/10/23 13:30:25 dk DISK01/d8/d16/f216 arset4.1 810d8.16cb
qfs2 119576.301 644930 t1/fileen f 0 0
A 2003/10/23 13:30:25 dk DISK01/d8/d16/f216 arset4.1 810d8.1bb8
qfs2 119577.301 1322899 t1/fileeo f 0 0
```

從左向右讀，以上清單中的欄位內容將顯示在表 3-1 中。

表 3-1 歸檔器記錄檔欄位

欄位	範例值	內容
1	A	歸檔活動如下所示： <ul style="list-style-type: none">• A 表示已歸檔。• R 表示已重新歸檔。• U 表示取消歸檔。
2	2001/03/23	以 <i>yyyy/mm/dd</i> 格式表示的歸檔動作日期。
3	18:42:06	以 <i>hh:mm:ss</i> 格式表示的歸檔作業時間。
4	mo	歸檔媒體類型。若需有關媒體類型的資訊，請參閱「mcf(4) 線上手冊」。
5	0004A	VSN。對可移除的媒體卡匣而言，此為磁碟區序列名稱。對磁碟歸檔而言，此為磁碟區名稱和歸檔 tar(1) 檔案路徑。

表 3-1 歸檔器記錄檔欄位 (續)

欄位	範例值	內容
6	arset0.1	歸檔組與備份數目。
7	9a089.1329	歸檔檔案在媒體 (tar(1) 檔案) 中的實體開始位置，以及在歸檔檔案中的檔案偏移 (以十六進制格式表示)。
8	samfs1	檔案系統名稱。
9	118.51	Inode 號碼與產生號碼。為表示唯一性，除了 inode 號碼之外，還會使用產生號碼；這是因為 inode 號碼可重複使用。
10	162514	檔案長度 (若檔案只寫在一個磁碟區上)。區段長度 (若檔案寫在多個磁碟區上)。
11	t0/fdn	相對於檔案系統之掛載點的檔案路徑與名稱。
12	f	檔案類型如下所示： <ul style="list-style-type: none"> • d 代表目錄。 • f 代表一般檔案。 • l 代表符號連結。 • R 代表可移除的媒體檔案。 • I 代表區段索引。 • S 代表資料區段。
13	0	溢位檔案或區段的部分。若此檔案為溢位檔案，此值則為非零。若為其他檔案類型，此值均為 0。
14	56	檔案歸檔所在之磁碟機的設備序號。

資料驗證

您可以視需要啟用歸檔副本的資料驗證。此功能會檢查複製到第二個和/或第三個媒體的任何資料上是否有資料毀壞。

資料驗證程序會執行寫入後讀取的驗證測試，並記錄此檔案之中介資料特性中資料有效性的確認。

您可以使用 `ssum` 選項，將檔案和目錄標記為需要驗證。子目錄會繼承其父系的資料驗證特性。您可以使用一般總和檢查方法，驗證寫入磁帶或磁碟歸檔的副本。

使用 `ssum -e` 指令設定檔案或目錄的資料驗證。如此會強制產生並使用歸檔和分段備份的總和檢查，並在建立所有歸檔副本並驗證其總和檢查之前，避免釋放檔案。僅有超級使用者可在檔案或目錄上設定此屬性。

備註 — 由於資料驗證請求會與一般分段備份請求一併置於分段備份程式佇列中，因此資料驗證功能會帶給分段備份程式資源額外的負擔。資料驗證也會導致額外的磁帶掛載，並因此可能影響歸檔器和分段備份程式的效能。由於在建立所有歸檔副本之前無法釋放檔案，因此使用資料驗證也可能需要更多的磁碟快取記憶體。

關於 archiver.cmd 檔案

archiver.cmd 檔案可控制歸檔器的運作方式。依據預設，每當啓動 sam-fsd 且掛載 Sun StorageTek SAM 檔案系統時，歸檔器就會執行。若無 archiver.cmd 檔案，歸檔器會採用下列預設值：

- 所有檔案皆會歸檔至所有可用的磁碟區。
- 所有檔案的歸檔時間為 4 分鐘。
- 歸檔間隔為 10 分鐘。

使用歸檔器指令檔 (archiver.cmd) 中指令可讓您自訂歸檔器的動作，以符合您站點的歸檔請求。

▼ 建立或修改 archiver.cmd 檔案及散佈您的變更

除了此方法以外，也可使用 File System Manager 軟體建立或修改 archiver.cmd 檔案。若需更多資訊，請參閱 File System Manager 線上說明。

1. (可選擇) 決定要編輯實際的 archiver.cmd 檔案，或是暫存的 archiver.cmd 檔案。

如果您有 /etc/opt/SUNwsamfs/archiver.cmd 檔案，而且您的系統已經在進行檔案的歸檔，則請執行此步驟。考慮將您的 archiver.cmd 檔案複製到一個可以在開始製作之前，進行編輯和測試的暫存位置。

2. 使用 vi(1) 或其他編輯器編輯檔案。

新增需要用來在您的站台控制歸檔的指令。若需有關您可以在此檔案中包括的指令之資訊，請參閱第 45 頁的「使用歸檔器指令」和第 82 頁的「關於磁碟歸檔」。

3. 儲存並關閉檔案。

4. 使用 archiver(1M) -lv 指令驗證檔案的正確性。

每當變更了 archiver.cmd 檔案，您應該使用 archiver(1M) 指令檢查語法錯誤。如下指定 archiver(1M) 指令，可針對目前的 Sun StorageTek SAM 系統，對 archiver.cmd 檔案進行評估：

```
# archiver -lv
```

此指令會產生所有選項的清單，並將 archiver.cmd 檔案、磁碟區、檔案系統內容及錯誤的清單寫入標準的輸出檔案 (stdout) 中。錯誤會阻止執行歸檔器。

預設 archiver(1M) 指令會評估 /etc/opt/SUNwsamfs/archiver.cmd 檔案中是否有錯誤。若要處理暫存的 archiver.cmd 檔案，請搭配使用 -c 選項與 archiver(1M) 指令，並提供此暫存檔的名稱。

5. 若發生錯誤，請在檔案中加以更正，然後重新執行 archiver(1M) 指令，以驗證更正是否已正確無誤。

您必須先更正所有錯誤，再繼續下一個步驟。如果歸檔器在 archiver.cmd 檔案中發現錯誤，則不會歸檔任何檔案。

6. 若是處理暫存檔，請將其移至 `/etc/opt/SUNWsamfs/archiver.cmd`。
7. 使用 `samd(1M) config` 指令散布檔案變更，然後重新啟動系統。

```
# samd config
```

archiver.cmd 檔案

archiver.cmd 檔案由以下類型的指令組成：

- 一般指令
- 歸檔組指派指令
- 歸檔組指令
- VSN pool 指令
- VSN 關聯指令

這些指令由從 archiver.cmd 檔案中讀取的文字行組成。每個指令行包含一個或多個由空格或 Tab 分隔的欄位。任何出現在井字號字元 (#) 後面的文字將被視為註解，並不作檢查。指令若是過長，可以在第一行的結尾加上反斜線 (\)，以延續到下一行。

archiver.cmd 檔案中的某些指令需要您指定時間單位或位元組單位。若要指定此類單位，請使用表 3-2 中的任一字母。

表 3-2 archiver.cmd 檔案指令單位

單位後綴	說明
時間後綴：	
s	秒
m	分
h	小時
d	天
w	週
y	年
大小後綴：	
b	位元組
k	KB
M	MB
G	GB
T	TB
P	PB
E	EB

archiver.cmd 檔案範例

程式碼範例 3-3 顯示 archiver.cmd 檔案範例。這些註釋表示各種指令類型。

程式碼範例 3-3 archiver.cmd 檔案範例

```
# Global directives

archivemeta = off
archmax = li 8G
examine = noscan
scanlist_squash = on

# limit the drives

drives = stk50 X

# File selection

fs = samfs1
logfile = /var/adm/samfs1.log
archive .
    1 -norelease 5m
    2 -norelease 5m
fs = samfs2
logfile = /var/adm/samfs2.log
archive .
    1 -norelease 5m
    2 -norelease 5m
fs = samfs3
logfile = /var/adm/samfs3.log
archive .
    1 -norelease 5m
    2 -norelease 5m
fs = samfs4
logfile = /var/adm/samfs4.log
archive .
    1 -norelease 5m
    2 -norelease 5m
fs = samfs5
logfile = /var/adm/samfs5.log
archive .
    1 -norelease 5m
    2 -norelease 5m
```

```
# The below information is for tape archiving.
# The recycler is not set up to actually recycle. It is set up for
# data checks and file recovery.

params
allsets -sort path -offline_copy stageahead -reserve set
allsets -recycle_ignore
allsets -recycle_hwm 50 -recycle_mingain 90 -recycle_vsncount 1
allsets.1 -startage 6h -startsize 8G -startcount 90000 -drives X -drivemin XXG
allsets.2 -startage 10h -startsize 12G -startcount 90000 -archmax 12G -drives
X -drivemin XXG
endparams

# Define VSNS for archive sets

vsns
archive.1 li .*
archive.2 li .*
endvsns
```

使用歸檔器指令

以下章節將解釋 archiver.cmd 指令。這些指令如下：

- [第 46 頁的「全域歸檔指令」](#)
- [第 55 頁的「檔案系統指令」](#)
- [第 56 頁的「歸檔組指定指令」](#)
- [第 63 頁的「歸檔備份指令」](#)
- [第 66 頁的「歸檔組備份參數」](#)
- [第 79 頁的「VSN 關聯指令」](#)
- [第 81 頁的「VSN 池指令」](#)

全域歸檔指令

全域指令可控制整體的歸檔器作業，並讓您最佳化站點配置的歸檔器作業。您可以直接將全域指令增加到 `archiver.cmd` 檔案中，或是使用 **File System Manager** 軟體進行指定。如需有關使用檔案系統管理設定全域指令的更多資訊，請參閱 **File System Manager** 線上說明。

`archiver.cmd` 檔案中的全域指令可透過第二欄位的等號 (=) 或不出現其他欄位的方式識別。

您必須先指定全域指令，才可在 `archiver.cmd` 檔案中指定 `fs=` 指令。`fs=` 是與特定檔案系統有關的指令。如果歸檔器在 `fs=` 指令後偵測到全域指令，它將會發出一則訊息。

archivemeta 指令：控制是否歸檔中介資料

`archivemeta` 指令可控制是否要對檔案系統中介資料進行歸檔。檔案若需經常移動，且檔案系統中的目錄結構也經常有所變更，便應對檔案系統中介資料進行歸檔。反之，目錄結構若是十分穩定，便可停用中介資料歸檔，以減少可移除式媒體磁碟機在載入或卸載卡匣時執行動作的次數。依據預設，會對中介資料進行歸檔。

此指令具有以下格式：

```
archivemeta=state
```

關於 `state`，您可以指定 `on` 或 `off`。預設值為 `on`。

中介資料歸檔程序會因您使用「版本 1」或「版本 2」超級區塊而有所不同，如下：

- 若是「版本 1」檔案系統，歸檔器會將目錄、可移除的媒體檔案、區段索引 `inode` 及符號連結歸檔為中介資料。
- 若是「版本 2」檔案系統，歸檔器便會將目錄與區段索引 `inode` 歸檔為中介資料。可移除的媒體檔案與符號連結會儲存在 `inode` 中，而非資料區段中。而且不會被歸檔。符號連結會歸檔為資料。

archmax 指令：控制歸檔檔案的大小

archmax 指令指定歸檔檔案的大小上限。使用者檔案組合在一起即形成歸檔檔案。當達到 *target-size* 值時，便不會再增加任何使用者檔案到歸檔檔案中。較大的使用者檔案將被寫入單一的歸檔檔案。

要變更預設值，請使用下列指令：

```
archmax=media target-size
```

表 3-3 archmax 指令的引數

引數	意義
<i>media</i>	表示媒體類型。如需有效媒體類型清單，請參閱「mcf(4) 線上手冊」。
<i>target-size</i>	歸檔檔案的大小上限。此值取決於媒體。根據預設值，寫入光碟的歸檔檔案不大於 5 MB。磁帶的最大預設歸檔檔案大小是 512 MB。

歸檔檔案大小設定為大或小，是各有利弊之處。例如，如果歸檔至磁帶，且將 archmax 設定為大，磁帶機將很少停止與啓動。但是，在寫入較大的歸檔檔案時，有可能會提前到達磁帶末端，從而會浪費較多的磁帶。一般而言，不應將 archmax 設定為超過媒體容量的 5%。

archmax 指令亦可為個別歸檔組設定。

備註 — archmax 指令對於將歸檔至 Sun StorageTek 5800 媒體類型的歸檔組而言，不是有效的指令。

bufsize 指令：設定歸檔器緩衝區大小

根據預設值，歸檔的檔案將被複製到使用記憶體緩衝區的歸檔媒體。您可以使用 `bufsize` 指令指定非預設的緩衝區大小 (選擇使用)，也可以鎖定緩衝區。這些動作可以提高效能，而您也可以嘗試使用不同的 `buffer-size` 值。

此指令具有以下格式：

```
bufsize=media buffer-size [lock]
```

表 3-4 `bufsize` 指令的引數

引數	意義
<code>media</code>	表示媒體類型。如需有效媒體類型清單，請參閱「 <code>mcf(4)</code> 線上手冊」。
<code>buffer-size</code>	2 至 1024 的數字。預設值為 4。此值會與該媒體類型的 <code>dev_blksize</code> 值相乘，並使用所得的緩衝區大小。 <code>dev_blksize</code> 值會在 <code>defaults.conf</code> 檔案中指定。如需有關此檔案的更多資訊，請參閱「 <code>defaults.conf(4)</code> 線上手冊」。
<code>lock</code>	<p>指出歸檔器在進行歸檔備份時，是否應使用鎖定的緩衝區。如果指定了 <code>lock</code>，歸檔器在 <code>sam-arcopy(1M)</code> 作業過程中會將檔案鎖定在記憶體的歸檔緩衝區內。如此可避免因鎖定及解除鎖定每個 I/O 請求的緩衝區所造成的經常性耗用時間，進而降低系統 CPU 時間。</p> <p><code>lock</code> 引數應該僅在具有大容量記憶體的大系統中指定。記憶體不足會導致出現記憶體不夠用的情況。</p> <p>僅在為要歸檔的檔案啟動了直接 I/O 時，<code>lock</code> 引數才會生效。依據預設，<code>lock</code> 參數並未指定，且檔案系統會鎖定所有直接 I/O 緩衝區，包括要歸檔的緩衝區。如需有關啟動直接 I/O 的更多資訊，請參閱「<code>setfa(1)</code> 線上手冊」、<code>sam_setfa(3)</code> 程式庫常式線上手冊，或「<code>mount_samfs(1M)</code> 線上手冊」上的 <code>-O forcedirectio</code> 選項。</p>

您可以使用 `-bufsize` 和 `-lock` 歸檔組複製參數，並以歸檔組為根據指定緩衝區大小及鎖定。如需更多資訊，請參閱第 66 頁的「歸檔組備份參數」。

drives 指令：控制用於歸檔的磁碟機數量

依據預設，歸檔器將使用自動程式庫中的所有磁碟機進行歸檔。若要限制使用的磁碟機數量，可使用 `drives` 指令。

此指令具有以下格式：

```
drives=auto-lib count
```

表 3-5 drives 指令的引數

引數	意義
<i>auto-lib</i>	如 <code>mcf</code> 檔案中所定義之自動程式庫的系列組名稱。
<i>count</i>	用於歸檔活動的磁碟機數目。

另請參閱第 68 頁的「指定用於歸檔請求的磁帶機數量：`-drivemax`、`-drivemin` 及 `-drives`」中所述的 `-drivemax`、`-drivemin` 及 `-drives` 歸檔組複製參數。

examine 指令：控制歸檔掃描

新檔案和已變更的檔案為歸檔的候選者。歸檔器會利用下列一種方式尋找這類檔案：

- 連續歸檔；使用此方式時，歸檔器會與檔案系統一起運作，以在檔案有所變更時能夠立即偵測到
- 掃描型歸檔；使用此方式時，歸檔器會定期掃描檔案系統，以尋找需要歸檔的檔案

此指令具有以下格式：

```
examine=method
```

對於 *method*，請指定表 3-6 中所顯示的任一關鍵字。

表 3-6 examine 指令之 *method* 引數的值

<i>method</i> 值	意義
noscan	指定連續歸檔。在初始掃描後，只會在內容變更和歸檔需要時才掃描目錄。不會掃描目錄和 inode 資訊。這種歸檔方法提供比掃描型歸檔更好的效能，特別是對有 1,000,000 檔案以上的檔案系統。預設值。
scan	指定掃描型歸檔。初始檔案系統掃描為目錄掃描。後續掃描為 inode 掃描。
scandirs	只在目錄上指定掃描型歸檔。當歸檔器發現設有 no_archive 屬性集的目錄時，將不會掃描該目錄。不會變更的檔案可置於此目錄，這種方式可以動態減低耗費於歸檔掃描的時間量。
scaninodes	只在 inode 上指定掃描型歸檔。

interval 指令：指定歸檔間隔

歸檔器會定期執行，以檢查所掛載之所有 Sun StorageTek SAM 檔案系統的狀態。執行時機由檔案系統上，每次掃描作業之間的歸檔間隔時間所控制。要變更時間，請使用 interval 指令。

只有在未設定連續歸檔，且未指定 startage、startsize 或 startcount 參數時，interval 指令才會啟動完整掃描。若設定了連續歸檔 (examine=noscan)，interval 指令便會以預設的 startage 值執行。

此指令具有以下格式：

```
interval=time
```

對於 *time*，請依照您的需求，指定檔案系統上每次掃描作業之間的時間。 *time* 的預設值為 600 (以秒為單位)，亦即 10 分鐘。如表 3-2 中所述，您可以指定其他時間單位，如分鐘或小時。

如果歸檔器接收到 samu(1M) 公用程式的 :arrun 指令，它將立即開始掃描所有檔案系統。如果也在 archiver.cmd 檔案中指定 examine=scan 指令，掃描則會在輸入 :arrun 或 :arscan 指令之後執行。

如果設定了檔案系統的 hwm_archive 掛載選項，歸檔間隔可自動減短。此掛載選項將指定在檔案系統裝滿且超過高標記時，歸檔器將開始掃描。high=*percent* 掛載選項會設定檔案系統的高標記。

如需有關指定歸檔間隔的更多資訊，請參閱「archiver.cmd(4) 線上手冊」。如需有關設定掛載選項的更多資訊，請參閱「mount_samfs(1M) 線上手冊」。

logfile 指令：指定歸檔器記錄檔

歸檔器可產生包含有關每個已歸檔、重新歸檔或自動取消歸檔的資訊之記錄檔。記錄檔是歸檔動作的連續記錄。要指定記錄檔，請使用 `logfile` 指令。

此指令具有以下格式：

```
logfile=pathname
```

關於 *pathname*，請指定絕對路徑和記錄檔名稱。根據預設值，將不產生此檔案。

`logfile` 指令亦可為個別檔案系統設定。

▼ 備份歸檔器記錄檔

假設您要將前一天的記錄檔複製到替代位置，以便每天備份歸檔器記錄檔。請務必在歸檔器記錄檔關閉時執行複製作業，而不要在記錄檔開啓進行寫入作業時執行。

1. 使用 `mv(1)` 指令可在 UNIX 檔案系統中移動歸檔器記錄檔。

這將會給予任何 `sam-arfind(1M)` 或 `sam-arcopy(1M)` 完成寫入歸檔器記錄檔的作業時間。

2. 使用 `mv(1)` 指令將前一天的歸檔器記錄檔移動到 Sun StorageTek SAM 檔案系統。

notify 指令：重新命名事件通知程序檔

`notify` 指令可設定歸檔器的事件通知程序檔之名稱。此指令具有以下格式：

```
notify=filename
```

關於 *filename*，請指定含有歸檔器事件通知程序檔或此檔案的完整路徑的檔案之名稱。

以下為預設的檔案名稱：

```
/etc/opt/SUNWsamfs/scripts/archiver.sh
```

歸檔器會以特定站台的方式執行此程序檔來處理各種事件。可使用下列任一關鍵字為第一個引數呼叫此程序檔：`emerg`、`alert`、`crit`、`err`、`warning`、`notice`、`info` 和 `debug`。

額外的引數在預設的指令碼中有說明。要取得更多相關資訊，請參閱「`archiver.sh(1M)` 線上手冊」。

ovflmin 指令：控制磁碟區溢位

藉由磁碟區溢位，歸檔檔案即可跨越多個磁碟區。在 `archiver.cmd` 檔案中使用 `ovflmin` 指令時，磁碟區溢位將會啟動。當檔案大小超出 `ovflmin` 指令的 `minimum-file-size` 引數值時，歸檔器便會將此檔案的一部分寫入其他可供使用且類型相同的磁碟區中。寫入每個磁碟區的部分檔案稱為一個區段。

備註 – 請僅在完全評估對站點的影響之後，再謹慎使用磁碟區溢位。災難復原與回收對於跨越磁碟區的檔案較為困難。如需更多資訊，請參閱「Sun StorageTek Storage Archive Manager 疑難排解指南」與「request(1) 線上手冊」。

歸檔器使用 `ovflmin` 指令控制磁碟區溢位。`ovflmin` 指令可指定觸發溢位程序的檔案大小臨界值。根據預設值，將停用磁碟區溢位。

此指令具有以下格式：

```
ovflmin = media minimum-file-size
```

表 3-7 `ovflmin` 指令的引數

引數	意義
<code>media</code>	表示媒體類型。如需有效媒體類型清單，請參閱「mcf(4) 線上手冊」。
<code>minimum-file-size</code>	要觸發磁碟區溢位的檔案大小下限。

假設有許多檔案佔用 `mo` 媒體極大的分段長度 (如 25%)。這些檔案在每個磁碟區上留下了未使用的空間。若要獲取較好的磁碟區空間壓縮，請將 `mo` 媒體的 `ovflmin` 設定為比最小檔案稍小的大小。下列指令將其設定為 150 MB：

```
ovflmin=mo 150m
```

請注意，在此範例中啟用磁碟區溢位亦會載入兩個磁碟區，以便能夠進行檔案的歸檔與分段備份；這是因為每個檔案皆會溢位至其他磁碟區。

`ovflmin` 指令亦可為個別歸檔組設定。

`sls(1)` 指令輸出會列出顯示各 VSN 上之檔案的各個區段的歸檔備份。[程式碼範例 3-4](#) 會顯示歸檔器記錄檔，而[程式碼範例 3-5](#) 則會顯示橫跨多磁碟區之大型檔案 `file50` 的 `sls -D` 指令輸出。

程式碼範例 3-4 歸檔器記錄檔範例

```
A 97/01/13 16:03:29 lt DLT000 big.1 7eed4.1 samfs1 13.7
477609472 00 big/file50 0 0
A 97/01/13 16:03:29 lt DLT001 big.1 7fb80.0 samfs1 13.7
516407296 01 big/file50 0 1
A 97/01/13 16:03:29 lt DLT005 big.1 7eb05.0 samfs1 13.7
505983404 02 big/file50 0 2
```

[程式碼範例 3-4](#) 顯示 `file50` 以 `DLT000`、`DLT001` 及 `DLT005` 的 VSN，橫跨三個磁碟區。每一個區段在磁碟區內的位置及大小，會分別顯示在第七與第十個欄位（第一個項目分別為 `7eed4.1` 與 `477609472`），並符合[程式碼範例 3-5](#) 中所顯示的 `sls -D` 輸出。有關歸檔器記錄項目的完整說明，請參閱「[archiver\(1M\)](#) 線上手冊」。

[程式碼範例 3-5](#) 顯示 `sls -D` 指令和輸出。

程式碼範例 3-5 `sls(1M) -D` 指令和輸出

```
# sls -D file50
file50:
mode: -rw-rw---- links: 1 owner: gmm group: sam
length: 1500000172 admin id: 7 inode: 1407.5
offline; archdone; stage -n
copy1: ---- Jan 13 15:55 lt
section 0: 477609472 7eed4.1 DLT000
section 1: 516407296 7fb80.0 DLT001
section 2: 505983404 7eb05.0 DLT005
access: Jan 13 17:08 modification: Jan 10 18:03
changed: Jan 10 18:12 attributes: Jan 13 16:34
creation: Jan 10 18:03 residence: Jan 13 17:08
```

磁碟區溢位檔案不會產生總和檢查。如需有關使用總和檢查的更多資訊，請參閱「[ssum\(1\)](#) 線上手冊」。

scanlist_squash 指令：控制掃描清單合併

scanlist_squash 參數可開啓或關閉掃描清單合併。預設設定為 off。此參數可針對全域或特定檔案系統。

開啓此選項時，有相同父目錄之兩個或兩個以上的子目錄，其檔案的掃描清單項目 (稍後會經由 sam-arfind 掃描) 會合併。此時若對許多子目錄皆有大幅變更的檔案系統進行歸檔，將會造成效能嚴重折損。合併掃描清單時，這些目錄會向上合併成共用的父目錄，導致深層遞迴掃描許多子目錄。

setarchdone 指令：控制 archdone 旗標的設定

setarchdone 參數是全域指令，可以控制 sam-arfind 檢查檔案時的 archdone 旗標設定。

此指令具有以下格式：

```
setarchdone=on|off
```

完成檔案的所有歸檔備份之後，檔案的 archdone 旗標會設定為表示無需採取其他歸檔動作。在進行 inode 掃描期間，歸檔器會偵測是否設定了 archdone 旗標；若已設定，歸檔器便不會再查閱 inode 的路徑名稱。

在目錄掃描期間，歸檔器亦會為永不歸檔的檔案設定 archdone 旗標。掃描大型目錄時，上述操作是耗時的作業，可能會影響效能。setarchdone 指令可讓您控制這項作業。若是將 examine 指令設定成 scandirs 或 noscan，則 setarchdone 指令的預設設定為 off。

此指令只能針對永不歸檔的檔案，控制其 archdone 旗標的設定。其不會影響歸檔備份完成之後的 archdone 旗標設定。

wait 指令：延遲歸檔器啓動

wait 指令會導致歸檔器等候來自 samu(1M) 或 File System Manager 的啓動訊號。依據預設，在由 sam-fsd(1M) 啓動後，歸檔器將開始歸檔。

此指令具有以下格式：

```
wait
```

wait 指令亦可為個別檔案系統設定。

檔案系統指令

您可以在 `archiver.cmd` 檔案中的一般指令之後，使用 `fs=` 指令併入特定檔案系統專用的指令。在遇到 `fs=` 指令之後，歸檔器會假設所有後續指令指定的動作都是僅限於對指定的檔案系統執行。

您可以編輯 `archiver.cmd` 檔案指定 `fs=` 指令 (如以下各節所述)，或是使用 **File System Manager** 軟體進行指定。如需更多資訊，請參閱 **File System Manager** 線上說明。

fs 指令：指定檔案系統

依據預設，歸檔控制適用於所有檔案系統。但是，您可對個別檔案系統進行某些限制。例如，您可使用此指令以便為每個檔案系統指定不同的記錄檔。要指定個別檔案系統，請使用 `fs` 指令。

此指令具有以下格式：

```
fs=fsname
```

關於 `fsname`，請指定在 `mcf` 檔案中定義的檔案系統名稱。

在這些指令後出現的一般指令與歸檔組關聯指令將僅套用於指定的檔案系統，直到出現其他的 `fs=` 指令。

全域與檔案系統指令

幾個指令可同時被指定為所有檔案系統的全域指令，與僅專用於一個檔案系統的指令。這些指令如下所示：

- `interval` 指令。如需有關此指令的更多資訊，請參閱第 50 頁的「[interval 指令：指定歸檔間隔](#)」。
- `logfile` 指令。如需有關此指令的更多資訊，請參閱第 51 頁的「[logfile 指令：指定歸檔器記錄檔](#)」。
- `scanlist_squash` 指令。如需有關此指令的更多資訊，請參閱第 54 頁的「[scanlist_squash 指令：控制掃描清單合併](#)」。
- `wait` 指令。如需有關此指令的更多資訊，請參閱第 54 頁的「[wait 指令：延遲歸檔器啟動](#)」。

歸檔組指定指令

依據預設，檔案會歸檔為針對該檔案系統所命名的歸檔組之一部分。但是，您可指定歸檔組以包括共享類似特性的檔案。如果檔案不符合其中一個指定的歸檔組，它將作為為檔案系統命名的預設歸檔組之部分進行歸檔。

您可以直接編輯 `archiver.cmd` 檔案建立歸檔組 (如以下各節所述)，也可以使用 **File System Manager** 軟體加以建立。在 **File System Manager** 中，*archive policy* 會定義歸檔組。若需更多資訊，請參閱 **File System Manager** 線上說明。

指定歸檔組

歸檔組成員身份指令會將具有類似特性的檔案指派給歸檔組。這些指令的語法將在 `find(1)` 指令後被模仿。每個歸檔組指派指令具有下列格式：

```
archive-set-name path [search-criterion1 search-criterion2 ... ] [file-attribute1 file-attribute2 ... ]
```

表 3-8 歸檔組指定指令的引數

引數	意義
<i>archive-set-name</i>	歸檔組的站點定義名稱。必須是歸檔組指派指令中的第一個欄位。歸檔組名稱通常表示屬於歸檔組的檔案特性。歸檔組名稱限制為字母、數字及底線字元 (<code>_</code>)。不允許使用其他特殊字元或空格。歸檔組名稱的第一個字元必須是字母。 為防止歸檔各種檔案，可將 <code>no_archive</code> 指定為 <i>archive-set-name</i> 值。
<i>path</i>	相對於檔案系統的掛載點之路徑這可允許讓歸檔組成員身份指令套用到多個 Sun StorageTek SAM 檔案系統。如果路徑要包括檔案系統中的所有檔案，可在路徑欄位中使用小數點 (<code>.</code>)。不允許在路徑中使用正斜線 (<code>/</code>)。 <i>path</i> 指定目錄中的檔案及其子目錄，將被視為包括在此歸檔組中。
<i>search-criterion1</i> <i>search-criterion2</i>	您可以不指定 <i>search-criterion</i> 引數，也可指定一或多個該引數。根據檔案大小、檔案所有權及其他因素，可指定搜尋條件以限制歸檔組。如需有關可用 <i>search-criterion</i> 引數的資訊，請參閱以下各節。
<i>file-attribute1</i> <i>file-attribute2</i>	您可以不指定 <i>file-attribute</i> 值，也可指定一或多個該項值。 <code>sam-arfind</code> 程序在歸檔過程中掃描檔案系統時，系統將為檔案設定這些檔案屬性。

程式碼範例 3-6 會顯示一般的歸檔組成員身份指令。

程式碼範例 3-6 歸檔組成員身份指令

```
hmk_files    net/home/hmk      -user hmk
datafiles    xray_group/data  -size 1M
system       .
```

您可以藉由將檔案包括於一個名為 `no_archive` 的歸檔組中，來抑制歸檔器。程式碼範例 3-7 顯示阻止進行 `tmp` 目錄 (任何層級) 中的檔案歸檔之文字行，而且無論 `tmp` 目錄位於該檔案系統的哪個目錄中。

程式碼範例 3-7 防止歸檔的歸檔指令

```
fs = samfs1
no_archive tmp
no_archive . -name .*/tmp/
```

以下章節將說明您可指定的 *search-criterion* 引數。

檔案時間 *search-criterion*：-access 和 -nftv

您可以使用 `-access age` 特性來指定一個檔案用來決定歸檔組成員身份的時間。當您使用此特性時，會將存取時間早於 `age` 的檔案重新歸檔到其他媒體。對於 `age`，請指定整數，並在其後指定表 3-9 中所列的任一個後綴。

表 3-9 -access age 後綴

後綴	意義
s	秒
m	分
h	小時
d	天
w	週
ly	年

例如，您可以使用此指令指定長時間未存取的檔案，使其重新歸檔到費用較低的媒體。

決定時間時，軟體會驗證檔案的存取及修改時間，以確定這些時間不僅大於或等於檔案的建立時間，同時亦小於或等於檢查檔案的時間。對已「遷移」到目錄中的檔案，此驗證可能無法產生期望的運作方式。`-nftv` (沒有檔案時間驗證) 參數可用於這些情形，以防止檔案存取與修改時間的驗證。

檔案時間 *search-criterion* : -after

您可以使用 `-after date-time` 特性，將最新修改或建立的檔案歸類到相同的歸檔組中。使用此特性時，只有在指定日期之後所建立或修改的檔案，才會納入歸檔組中。

date-time 的格式為 `YYYY-MM-DD[Thh:mm:ss] [Z]` (ISO 8601 格式)。若未指定時間部分，便會將其假設為 `00:00:00`。如有 `Z`，便會將時間視為「格林威治標準時間 (UTC)」；否則將視為本機時間。

檔案大小 *search-criterion* : -minsize 和 -maxsize

透過 `-minsize size` 和 `-maxsize size` 特性，可以使用檔案大小決定歸檔組的成員身份。關於 *size*，請指定一個整數，後面接著表 3-10 顯示的其中一個字母。

表 3-10 `-minsize` 和 `-maxsize size` 後綴

字母	意義
b	位元組
k	KB
M	MB
G	GB
T	TB
P	PB
E	EB

範例：程式碼範例 3-8 中的文字行可指定將所有大於 500 KB，但小於 100 MB 的檔案歸類於歸檔組 `big_files`。大於 100 MB 的檔案則歸類於歸檔組 `huge_files`。

程式碼範例 3-8 使用 `-minsize` 和 `-maxsize` 指令範例

```
big_files . -minsize 500k -maxsize 100M
huge_files . -minsize 100M
```

所有者與群組 *search-criterion* : `-user` 和 `-group`

合併使用所有權與群組，即可透過 `-user name` 和 `-group name` 特性決定歸檔組的成員身份。在程式碼範例 3-9 中，所有屬於使用者 `sysadmin` 的檔案，皆屬於歸檔組 `adm_set`，而所有具有 `marketing` 群組名稱的檔案，則會位在歸檔組 `mktng_set` 中。

程式碼範例 3-9 `-user` 與 `-group` 指令的使用範例

```
adm_set      . -user sysadmin
mktng_set    . -group marketing
```

使用式樣比對的檔案名稱 *search-criterion* : `-name regex`

要包括在歸檔組內的檔案名稱，可透過常規表示式加以指定。將 *search-criterion* 指令指定為 `-name regex`，可指定任何符合常規表示式 *regex* 之完整路徑者，皆為歸檔組成員。

regex 引數會遵循「`regexp(5)` 線上手冊」中所列的慣例。請注意，常規表示式與 UNIX 萬用字元遵循不同的規則。

所有位於所選目錄下的檔案 (包括其相對於檔案系統掛載點的指定路徑) 皆會進行式樣比對。這將允許您在 `-name regex` 欄位中建立模式，以符合檔案名稱與路徑名稱。

範例

以下指令會將歸檔組 `images` 中的檔案限制為以 `.gif` 結尾的那些檔案：

```
images . -name \.gif$
```

以下指令將選擇以字元 `GEO` 開頭的檔案：

```
satellite . -name /GEO
```

您可使用具有 `no_archive` 歸檔組的常規表示式。以下指定可防止歸檔任何以 `.o` 結尾的檔案：

```
no_archive . -name \.o$
```

假設您的 archiver.cmd 檔案含有程式碼範例 3-10 中所示的文字行。

程式碼範例 3-10 常規表示式檔案

```
# File selections.
fs = samfs1
    1 1s
    2 1s
no_archive share/marketing -name fred\.
```

透過此 archiver.cmd 檔案，歸檔器即不會對使用者目錄或子目錄中的 fred.* 進行歸檔。程式碼範例 3-11 會顯示您指定程式碼範例 3-10 所示之指令時所不會進行歸檔的檔案。

程式碼範例 3-11 不歸檔的檔案 (使用程式碼範例 3-10 中所示的指令)

```
/sam1/share/marketing/fred.anything
/sam1/share/marketing/first_user/fred.anything
/sam1/share/marketing/first_user/first_user_sub/fred.anything
```

程式碼範例 3-12 顯示已歸檔的檔案 (如果您指定程式碼範例 3-10 中所示的指令)。

程式碼範例 3-12 已歸檔的檔案 (使用程式碼範例 3-10 中所示的指令)

```
/sam1/fred.anything
/sam1/share/fred.anything
/sam1/testdir/fred.anything
/sam1/testdir/share/fred.anything
/sam1/testdir/share/marketing/fred.anything
/sam1/testdir/share/marketing/second_user/fred.anything
```

相對於程式碼範例 3-10，假設您的 archiver.cmd 檔案含有程式碼範例 3-13 中所示的文字行。

程式碼範例 3-13 archiver.cmd 檔案範例

```
# File selections.
fs = samfs1
    1 1s
    2 1s
no_archive share/marketing -name ^share/marketing/[^/]*fred\.
```

程式碼範例 3-13 中的 archiver.cmd 檔案將不歸檔使用者主目錄中的 fred.*，但將歸檔使用者子目錄與目錄 share/marketing 中的 fred.*。在此案例中，所有介於 share/marketing/ 至下一個斜線字元 (/) 之間的事物，皆會被視為使用者主目錄。因此，將不會對下列檔案進行歸檔：

```
/sam1/share/marketing/first_user/fred.anything
```

程式碼範例 3-14 顯示已歸檔的檔案 (如果您指定程式碼範例 3-13 中所示的指令)。

程式碼範例 3-14 已歸檔的檔案 (使用程式碼範例 3-13 中所示的指令)

```
/sam1/share/fred.anything  
/sam1/share/marketing/fred.anything  
/sam1/share/marketing/first_user/first_user_sub/fred.anything  
/sam1/fred.anything  
/sam1/testdir/fred.anything  
/sam1/testdir/share/fred.anything  
/sam1/testdir/share/marketing/fred.anything  
/sam1/testdir/share/marketing/second_user/fred.anything  
/sam1/testdir/share/marketing/second_user/sec_user_sub/fred.any
```

釋放及分段備份 *file-attributes* : -release 和 -stage

您可以使用 -release 和 -stage 選項，個別設定與歸檔組內的檔案相關之釋放和分段備份屬性。這些設定值將覆寫使用者之前可能已設定的分段備份或釋放屬性。

-release 選項具有以下格式：

```
-release attribute
```

-release 指令的屬性與 release(1) 指令遵循相同的慣例，如表 3-11 所示。

表 3-11 -release 指令的屬性

屬性	意義
a	釋放完成第一個歸檔備份後的檔案。
d	重設回預設值。
n	永不釋放檔案。
p	部分釋放檔案的磁碟空間。

-stage 選項具有以下格式：

```
-stage attribute
```

-stage 指令的屬性與 stage(1) 指令遵循相同的慣例，如表 3-12 所示。

表 3-12 -stage 指令的屬性

屬性	意義
a	以關聯方式分段備份此歸檔組中的檔案。
d	重設回預設值。
n	永不分段備份歸檔組中的檔案。

下列範例將顯示如何使用檔案名稱指定與檔案屬性釋放部分的 Macintosh 資源目錄：

```
MACS . -name .*/\ .rscs/ -release p
```

歸檔組成員身份衝突

有時為包括在歸檔組中的檔案選擇路徑與其他檔案特性，會導致不明確的歸檔組成員身份。這些情況可按下列方式解決：

1. 將選擇首先出現在歸檔組中的成員身份定義。
2. 在任何全域定義之前，首先選擇本機於檔案系統的成員身份定義。
3. 完全重複前一個定義的成員身份定義被記錄為錯誤。

使用這些規則時，應事先在指令檔中指定更具限制性的成員身份定義。

控制特定檔案系統 (使用 `fs=fsname` 指令) 的歸檔時，歸檔器會先評估檔案系統專用指令，再評估全域指令。因此，檔案可被指定給本機歸檔組 (包括 `no_archive` 歸檔組)，而不是被指定給全域歸檔。這對 `no_archive` 一類的全域歸檔組指定將會造成影響。

在程式碼範例 3-15 中，管理員似乎不想對兩個檔案系統上的任何 .o 檔案進行歸檔。但是，因為在全域歸檔組指定 no_archive 之前，本機歸檔組指定 allfiles 已被評估，samfs1 與 samfs2 檔案系統中的 .o 檔案將被歸檔。

程式碼範例 3-15 可能具有成員身份衝突的 archiver.cmd 檔案

```
no_archive . -name *.*\.*$
fs = samfs1
    allfiles .
fs = samfs2
    allfiles .
```

程式碼範例 3-16 所顯示的指令，可用以確保不會對兩個檔案系統上的任何 .o 檔案進行歸檔。

程式碼範例 3-16 更正的 archiver.cmd 檔案

```
fs = samfs1
    no_archive . -name *.*\.*$
    allfiles .
fs = samfs2
    no_archive . -name *.*\.*$
    allfiles .
```

歸檔備份指令

若不指定歸檔備份，歸檔器將為歸檔組中的檔案寫入單一歸檔備份。根據預設值，將在檔案的歸檔時間為四分鐘時製作此備份。如需多個歸檔備份，必須使用歸檔備份指令指定包括第一個備份在內的所有備份。

歸檔備份指令會以 *copy-number* 值 1、2、3 或 4 開頭。數字後面是為此備份指定歸檔特性的一個或多個引數。

歸檔備份指令必須緊接在與其相關的歸檔組指定指令之後。每個歸檔備份指令具有下列格式：

```
copy-number [ -release | -norelease ] [archive-age] [unarchive-age]
```

您可以編輯 archiver.cmd 檔案指定歸檔備份指令 (如此處所述)，或是使用 File System Manager 軟體進行指定。若需更多資訊，請參閱 File System Manager 線上說明。

下列章節將描述歸檔備份指令引數。

歸檔後釋放磁碟空間：-release

若要指定在完成歸檔備份之後自動釋放檔案所需的磁碟空間，請在備份數目後使用 `-release` 指令。此指令具有以下格式：

```
-release
```

在**程式碼範例 3-17** 中，具有群組 `images` 的檔案在其歸檔時間達到 10 分鐘時將被歸檔。在製作歸檔備份 1 後，磁碟快取記憶體空間將釋放。

程式碼範例 3-17 使用 `-release` 指令的 `archiver.cmd` 檔案

```
ex_set . -group images
      1 -release 10m
```

延遲釋放磁碟空間：-norelease

您可能直到完成多個歸檔備份才想釋放磁碟空間。`-norelease` 選項可防止自動釋放磁碟快取記憶體空間，直到製作所有標記 `-norelease` 的備份。

此指令具有以下格式：

```
-norelease
```

`-norelease` 指令可以指定歸檔組在所有備份完成歸檔之後釋放，但必須先呼叫釋放器，並將檔案選為釋放候選者，才可釋放這些檔案。

程式碼範例 3-18 指定名為 `vault_tapes` 的歸檔組。將建立兩個備份，但是與此歸檔組相關的磁碟快取記憶體，將直到製作兩個備份才被釋放。

程式碼範例 3-18 使用 `-norelease` 指令的 `archiver.cmd` 檔案

```
vault_tapes
  1 -norelease 10m
  2 -norelease 30d
```

在單一副本上使用 `-norelease` 指令，不會影響自動釋放。

同時使用 `-release` 和 `-norelease`

若要確保磁碟空間可以在所有歸檔組備份完成歸檔之後立即釋放，可以同時使用 `-release` 和 `-norelease` 選項。合併使用 `-release` 與 `-norelease` 可以讓歸檔器在所有具有此合併項目的備份完成之後，立即釋放磁碟空間，而不像單獨使用 `-norelease` 選項般，必須在呼叫釋放器之後才能夠予以釋放。

設定歸檔時間

您可以在歸檔備份指令中指定歸檔時間，以設定檔案的歸檔時間。歸檔時間可使用 `h` (代表小時) 或 `m` (代表分鐘) 等後綴字元進行指定，如表 3-2 中所示。

在程式碼範例 3-19 中，`data` 目錄中的檔案在其歸檔時間達到一小時的時候將被歸檔。

程式碼範例 3-19 指定歸檔時間的 `archiver.cmd` 檔案

```
ex_set data
  1 1h
```

自動取消歸檔

如果您定檔案的多個歸檔備份，則可能自動取消歸檔除了一個之外的所有備份。使用不同歸檔時間將檔案歸檔至各種媒體時，可能需要執行此作業。

程式碼範例 3-20 會顯示用以指定取消歸檔時間的指令。路徑 `home/users` 中檔案的第一個備份，將在修改後六分鐘被歸檔。在檔案達到 10 週時，將製作第二與第三個歸檔備份。然後將回復第一個備份。

程式碼範例 3-20 指定取消歸檔時間的 `archiver.cmd` 檔案

```
ex_set home/users
  1 6m 10w
  2 10w
  3 10w
```

若需有關控制取消歸檔的更多方法，請參閱第 73 頁的「[控制取消歸檔](#)」。

為中介資料指定多個備份

如果需要一個以上的中介資料備份，您可以在 `fs=` 指令之後，立即將備份定義置於程序檔案中。

[程式碼範例 3-21](#) 顯示需要指定多個中介資料備份的 `archiver.cmd` 檔案。

程式碼範例 3-21 指定多個中介資料備份的 `archiver.cmd` 檔案

```
fs = samfs7
    1 4h
    2 12h
```

在此範例中，將會在 4 小時之後製作 `samfs7` 檔案系統之中介資料的第一個備份，接著再於 12 小時之後製作第二個備份。

檔案系統中介資料包括檔案系統中的路徑名稱。因此，您若是經常對目錄進行變更，將會根據新的路徑名稱而建立新的歸檔備份。這將導致經常載入為中介資料指定的磁碟區。

歸檔組備份參數

`archiver.cmd` 檔案的歸檔組參數區段是以 `params` 指令做為開頭，以 `endparams` 指令做為結尾。[程式碼範例 3-22](#) 顯示歸檔組指令的格式。

程式碼範例 3-22 歸檔組備份參數格式

```
params
archive-set-name.copy-number[R] [ -param1 -param2 ...]
.
.
.
endparams
```

表 3-13 歸檔組備份參數的引數

引數	意義
<i>archive-set-name</i>	歸檔組的站點定義名稱。通常是表示屬於該歸檔組的檔案特性。可以是 <code>allsets</code> 。歸檔組名稱限制為字母、數字及底線字元 (<code>_</code>)。不允許使用其他特殊字元或空格。歸檔組名稱的第一個字元必須是字母。
.	小數點號 (<code>.</code>) 字元。用以分隔 <i>archive-set-name</i> 與 <i>copy-number</i> 。
<i>copy-number</i>	定義歸檔備份數量的整數。可以是 1、2、3 或 4。
R	指定被定義的參數是用於此歸檔組的重新歸檔備份。例如，您可以使用 R 並指定 <i>-param1</i> 引數中的 VSN，以將重新歸檔備份導向特定磁碟區。
<i>-param1</i> <i>-param2</i>	一或多個參數。以下子區段描述可在 <code>params</code> 和 <code>endparams</code> 指令之間指定的參數。

您可以編輯 `archiver.cmd` 檔案指定歸檔組備份參數 (如此處所示)，或是使用 File System Manager 軟體進行指定。若需更多資訊，請參閱 File System Manager 線上說明。

虛擬歸檔組 `allsets` 可提供為所有歸檔組設定預設歸檔組目錄的方法。所有 `allsets` 指令皆須位在實際的歸檔組備份指令之前。個別歸檔組備份的設定參數將覆寫 `allsets` 指令設定的參數。如需有關 `allsets` 歸檔組的更多資訊，請參閱「`archiver.cmd(4)` 線上手冊」。

以下各節將說明磁碟歸檔參數以外的所有歸檔組處理參數。若需有關磁碟歸檔參數的資訊，請參閱第 82 頁的「關於磁碟歸檔」。

控制歸檔檔案的大小：`-archmax`

`-archmax` 指令可設定歸檔組的檔案大小上限。此指令具有以下格式：

```
-archmax target-size
```

此指令相當近似於 `archmax` 全域指令。如需有關該指令與 *target-size* 輸入值的資訊，請參閱第 47 頁的「`archmax` 指令：控制歸檔檔案的大小」。

設定歸檔器緩衝區大小：-bufsize

預設在將歸檔檔案寫入歸檔媒體之前，會將此檔案儲存在緩衝區的記憶體中。您可以使用 `-bufsize` 指令指定非預設的緩衝區大小。這些動作可以提高效能，而您也可以嘗試使用不同的 `buffer-size` 值。

此參數具有下列格式：

```
-bufsize=buffer-size
```

對於 `buffer_size`，請指定 2 至 32 的數字。預設值為 4。此值會與該媒體類型的 `dev_blksize` 值相乘，並使用所得的緩衝區大小。`dev_blksize` 值會在 `defaults.conf` 檔案中指定。如需有關此檔案的更多資訊，請參閱「`defaults.conf(4)` 線上手冊」。

例如，此參數可在 `archiver.cmd` 檔案的以下文字行中指定：

```
myset.1 -bufsize=6
```

在全域範圍中，此指令的等效項目為 `bufsize=media buffer-size`。如需有關該指令的更多資訊，請參閱第 48 頁的「`bufsize` 指令：設定歸檔器緩衝區大小」。

指定用於歸檔請求的磁帶機數量：-drivemax、-drivemin 及 -drives

預設歸檔器只會使用一個媒體磁碟機歸檔一個歸檔組中的檔案。當歸檔組中具有許多檔案或大型檔案時，使用多部磁碟機可能較為有利。此外，您自動程式庫中的磁碟機運作速度若各有不同，可使用多部磁碟機平衡這些差異，進而提升歸檔效能。

此磁碟機指令具有下列格式：

```
-drivemax max-size  
-drivemin min-size  
-drives number
```

表 3-14 -drivemax、-drivemin 及 -drives 指令的引數

引數	意義
<i>max-size</i>	要使用一個磁帶機進行歸檔的資料最大數量。
<i>min-size</i>	要使用一個磁帶機進行歸檔的資料最小數量。預設值為 <code>-archmax target-size</code> 值 (如有指定) 或該媒體類型的預設值。 您若是指定 <code>-drivemin min-size</code> 指令，Sun StorageTek SAM 軟體便只會在具備足夠工作量的情況下才會使用多部磁碟機。其準則是將 <code>min-size</code> 儘量設大，以使傳輸時間明顯長於卡匣更換時間 (載入、定位、卸載)。
<i>number</i>	要在歸檔此歸檔組時使用的磁碟機數量。預設值是 1。

歸檔請求會針對指定的參數加以評估，詳情如下：

- 歸檔請求若小於 *min-size* 的值，便只會使用一部磁碟機寫入歸檔請求。
- 歸檔請求若大於 *min-size* 的值，將會根據 *min-size* 評估歸檔請求，而磁碟機的適當數量最多則可排定為所指定的全部磁碟機數量。
- 若 *min-size* 的值為 0，便會嘗試在指定之所有數量的磁碟機上分割歸檔請求。

在使用 `-drives` 參數時，僅在要歸檔的資料超出 *min-size* 的值時，才會使用多部磁碟機。下列二值中的較小者，即是要同時使用的磁碟機數量：

- *min-size* 值所切分的歸檔請求大小
- `-drives` 參數所指定的磁碟機數量

若要切分磁碟機的歸檔請求，但不希望造成所有磁碟機因為小型歸檔請求而效能降低，可以使用 `-drivemin` 和 `-drives` 參數。這可能套用於使用較大檔案的操作。

若要設定這些參數，必須考量檔案建立速率、磁碟機數量、載入和卸載磁碟機所需的時間及磁碟機傳輸速率。

例如，假設您要在五部磁碟機之間分割名為 `bigfiles` 的歸檔組。根據其大小，此歸檔組可按表 3-15 中所示進行分割。

表 3-15 歸檔組範例分割

歸檔組大小	磁帶機數目
< 20 GB	1
> 20 GB 至 < 30 GB	2
> 30 GB 至 < 40 GB	3
> 40 GB 至 < 50 GB	4
> 50 GB	5

程式碼範例 3-23 顯示 `archiver.cmd` 檔案中用來分割多個磁帶機上的歸檔請求的指令行。

程式碼範例 3-23 用來分割多個磁帶機上的歸檔請求的指令

```
params
bigfiles.1 -drives 5 -drivemin 10G
endparams
```

此外，您也可以在此 `archiver.cmd` 檔案中指定下行：

```
huge_files.2 -drives 2
```

在歸檔組 `huge_files.2` 中的檔案總大小等於或大於媒體 `drivemin` 的兩倍時，將有兩個磁帶機可用於歸檔檔案。

最大化磁碟區上的空間：-fillvsns

歸檔器預設會在寫入歸檔備份時，從指定給歸檔組之所有磁碟區中，選取具備足夠空間供所有檔案使用的磁碟區。此動作可讓磁碟區不至於到達容量上限。如果指定了 -fillvsns，歸檔器會將歸檔請求分隔為數個較小的群組。

指定歸檔緩衝區鎖定：-lock

預設在將歸檔檔案寫入歸檔媒體之前，會將此檔案儲存在緩衝區的記憶體中。如果已啟動直接 I/O，您可使用 -lock 參數鎖定此緩衝區。此動作可提升效能。

此參數具有下列格式：

```
-lock
```

-lock 參數會指定歸檔器應使用鎖定的緩衝區製作歸檔備份。如果指定了 -lock，歸檔器在 sam-arcopy(1M) 作業過程中會將檔案鎖定在記憶體的歸檔緩衝區內。如此可避免分頁緩衝區，進而提高效能。

-lock 參數應該僅在具有大容量記憶體的大系統中指定。記憶體不足會導致出現記憶體不夠用的情況。

僅在為要歸檔的檔案啟動了直接 I/O 時，-lock 參數才會生效。預設不會指定 -lock 參數，且檔案系統會鎖定包括要歸檔之緩衝區在內的所有直接 I/O 緩衝區。如需有關啟動直接 I/O 的更多資訊，請參閱「setfa(1) 線上手冊」、sam_setfa(3) 程式庫常式線上手冊，或「mount_samfs(1M) 線上手冊」上的 -O forcedirectio 選項。

例如，此參數可在 archiver.cmd 檔案的以下文字行中指定：

```
yourset.3 -lock
```

您亦可指定 bufsize=*media buffer-size* [lock] 指令的 lock 引數，以在全域範圍內指定此參數的等效項目。如需有關此主題的更多資訊，請參閱第 48 頁的「bufsize 指令：設定歸檔器緩衝區大小」。

製作離線檔案的歸檔備份：-offline_copy

檔案為在一個歸檔備份製作完畢後，要被釋放的候選者。如果檔案在所有歸檔備份製作完畢之前就釋放並離線，歸檔器會使用此參數決定在製作其他歸檔備份時要使用的方法。選擇所要使用的方法時，必須考量 Sun StorageTek SAM 系統可用的磁碟機數目，以及可用的磁碟快取記憶體總量。

此參數具有下列格式：

```
-offline_copy method
```

對於 *method*，請指定表 3-16 中所顯示的任一關鍵字。

表 3-16 -offline_copy 指令之 *method* 引數的值

<i>method</i> 值	意義
none	在複製到歸檔磁碟區之前，視各個檔案的需要分段備份檔案。預設值。
direct	直接將檔案從離線磁碟區複製到歸檔磁碟區，而不使用快取。此方法假設來源磁碟區和目標磁碟區是不同的磁碟區，而且兩個磁帶機皆可用。如果指定了這種方法，請將 <code>stage_n_window</code> 掛載選項的值提高到大於其預設 256 KB 的值。如需有關掛載選項的更多資訊，請參閱「 <code>mount_samfs(1M)</code> 線上手冊」。
stageahead	在歸檔另一個檔案的同時，分段備份一個檔案。系統在將檔案寫入其目標時，即會開始對下一個歸檔檔案進行分段備份。
stageall	在進行歸檔之前，將所有檔案分段備份至磁碟快取記憶體。此方法只使用一個磁帶機，並假設磁碟快取記憶體上的空間足以容納所有檔案。

指定回收

回收程序可讓您收回過期歸檔影像所佔用的歸檔磁碟區空間。依據預設，不會產生回收。

若要進行回收，可以在 `archiver.cmd` 檔案和 `recycler.cmd` 檔案中指定指令。如需有關 `archiver.cmd` 檔案支援的回收指令之更多資訊，請參閱第 133 頁的「回收」。

關聯歸檔：-join path

若指定了 `-join path` 參數，歸檔器便會使用關聯歸檔。如果要將整個目錄檔至一個磁碟區，而且您知道歸檔檔案實際僅可位於一個磁碟區上，關聯歸檔則很實用。否則，如果您要將目錄保留在一起，可使用 `-sort path` 或 `-rsort path` 參數以保持檔案的連續性。`-rsort` 參數可指定反向排序。

在歸檔器將歸檔檔案寫入磁碟區時，它會有效率地將使用者檔案納入磁碟區中。之後在存取相同目錄中的檔案時，將會因為分段備份程序在磁碟區中移動，藉以讀取下一個檔案而發生延遲。若要減少延遲，可使用 `-join path` 參數對歸檔組備份中，位於相同目錄路徑下的檔案進行連續歸檔。關聯歸檔程序將覆寫空間效率演算法，以將相同目錄中的檔案歸檔在一起。

檔案內容若無變更，但隨時皆要以整組的方式存取某個檔案群組，即可使用關聯歸檔。以醫院為例，您若要存取某位患者所有關聯的醫療影像，即可使用關聯歸檔。例如：

```
patient_images.1 -join path
```

備註 - `-join path` 參數會將相同目錄中的資料檔寫入相同的歸檔檔案。如果有許多包含小檔案的目錄，歸檔器將建立許多小歸檔檔案。這些小型且不連續的歸檔檔案各有其 `tar(1)` 標頭，因此會減緩系統的寫入效能。

此外，由於 `-join path` 參數會指定將所有位在相同目錄下的檔案，歸檔到同一磁碟區上，因此可能會發生檔案群組無法容納於任何可用的磁碟區中。在此情況下，檔案直到將更多的磁碟區指派給歸檔組才被歸檔。亦可能會出現要歸檔的檔案群組太大，無法符合單一磁碟區的情況。在此情況下，檔案將永遠無法歸檔。

對於大多數的應用程式而言，若執行 `-join path` 這類較具限制性的作業並非必要，建議使用 `-sort path` 或 `-join path` 參數。

亦可能按時間、大小或路徑將歸檔組備份中的檔案排序。`age` 和 `size` 引數不能同時使用。[程式碼範例 3-24](#) 顯示如何使用 `-sort` 參數加上 `age` 或 `size` 引數，來排序歸檔組。

程式碼範例 3-24 排序歸檔組的指令

```
cardiac.1 -sort path
cardiac.2 -sort age
catscans.3 -sort size
```

第一行會強制歸檔器按路徑名稱將歸檔請求排序。第二個範例行會強制歸檔器依照檔案的時間，以最舊到最新的方式排序歸檔組備份 `cardiac.2`。第三行會強制依照檔案的大小，以最小至最大的方式排序歸檔組備份 `catscans`。若要使用反向排序，可以指定 `-rsort` 取代 `-sort`。

控制取消歸檔

取消歸檔是指刪除檔案或目錄之歸檔項目的程序。檔案將根據上次存取的時間被取消歸檔。所有需要經常存取的資料可以儲存在快速媒體 (如磁碟) 上；而較舊、較少存取的資料則可儲存在磁帶上。根據預設值，檔案將永不取消歸檔。

例如，假設程式碼範例 3-25 中所顯示的 `archiver.cmd` 檔案負責控制經常存取的檔案。此檔案會一直保存在磁碟上，即使留存時間超過了 60 天亦然。只要檔案在 60 天內沒有被存取，備份 1 資訊將被移除。

如果備份 1 資訊被移除 (因為 60 天沒有存取檔案)，且某人從備份 2 中分段備份檔案，它將被從磁帶中讀取。在檔案回到線上狀態時，歸檔器將在磁碟上製作新備份 1，並且 60 天的存取循環將重新啟動。如果再次存取檔案，Sun StorageTek SAM 歸檔器將重新產生備份 1。

程式碼範例 3-25 控制取消歸檔的指令

```
arset1 dir1
  1    10m    60d
  2    10m
  3    10m
vsns
arset1.1 mo    OPT00[0-9]
arset1.2 lt    DLTA0[0-9]
arset1.3 lt    DLTB0[0-9]
```

假設患者在醫院四週。在此期間，此患者的所有檔案將儲存在快速媒體上 (備份 1=mo)。四週後，患者將出院。如果在此患者出院後 60 天沒有人存取其資料，`inode` 中的備份 1 記錄將被取消歸檔，並且僅有備份 2 與備份 3 記錄可用。磁碟區現在即被回收，以便為目前更多的患者留出空間，而無需增加程式庫。患者若在六個月後返回醫院進行追蹤檢查，第一次將會從磁帶 (備份 2) 存取資料。此時歸檔器會自動在磁碟上建立新備份 1，以確保追蹤檢查期間 (需要幾天或幾週)，可以從快速媒體上獲得資料。

控制歸檔檔案的寫入方式：-tapenonstop

依據預設，歸檔器可在歸檔檔案之間寫入一個磁帶標記、一個檔案結尾 (EOF) 標籤以及再兩個磁帶標記。在啟動下一個歸檔檔案時，磁帶機將回到導致效能喪失的第一個磁帶標記後之位置。`-tapenonstop` 參數將指示歸檔器僅寫入初始的磁帶標記。此外，如果指定 `-tapenonstop` 參數，歸檔器將在備份作業結束時輸入歸檔資訊。

如需有關 `-tapenonstop` 參數的更多資訊，請參閱「`archiver.cmd(4)` 線上手冊」。

預留磁碟區：-reserve

依據預設，歸檔器會將歸檔組備份寫入任何由常規表示式指定的磁碟區，該常規表示式在 `archiver.cmd` 檔案的磁碟區關聯部分有說明。但您有時可能會希望歸檔組磁碟區中只含有單一歸檔組中的檔案。您可以保留磁碟區，以滿足這項資料儲存需求。

`-reserve` 參數可為歸檔組預留磁碟區。在設定 `-reserve` 參數且將磁碟區指定歸檔組備份之後，歸檔識別標記將不會被指定給任何其他歸檔組備份，即使有常規表示式與其相符。

備註 – 使用預留磁碟區的站台可能會引起更多的卡匣載入與卸載。

選取歸檔組所要使用的磁碟區時，會將該磁碟區指定以保留的名稱，亦即連結了歸檔組與磁碟區的唯一識別碼。

備註 – `-reserve` 參數計劃預留由一個歸檔組單獨使用的磁碟區。許多包含小檔案的目錄，會導致許多小歸檔檔案被寫入每個預留的磁碟區。這些小型且不連續的歸檔檔案各有其 `tar(1)` 標頭，因此會減緩系統的效能。

`-reserve` 參數的格式如下所示：

```
-reserve keyword
```

keyword 值取決於您所使用的表單，如下：

- **歸檔組表單** – 如表 3-17 所示，`set` 關鍵字可指定保留名稱中的歸檔組元件。

表 3-17 歸檔組表單範例

指令與關鍵字	預留名稱範例
<code>-reserve set</code>	<code>users.1//</code>
	<code>Data.1//</code>

以程式碼範例 3-26 中的 `archiver.cmd` 檔案片段為例，其會顯示以 `allsets` 歸檔組名稱為開頭的該行，其會依據歸檔組為所有的歸檔組保留磁碟區。

程式碼範例 3-26 按歸檔組預留磁碟區

```
params
allsets -reserve set
endparams
```

- **所有者表單** – `dir`、`user` 與 `group` 等彼此互斥的關鍵字，可指定保留名稱中的所有者元件。`dir` 關鍵字會使用緊接在歸檔組定義路徑指定之後的目錄路徑元件。[表 3-18](#) 會顯示相關範例。

表 3-18 所有者組表單範例

指令與關鍵字	預留名稱範例
<code>-reserve dir</code>	<code>proj.1/p105/</code> <code>proj.1/p104/</code>
<code>-reserve user</code>	<code>users.1/user5/</code> <code>users.1/user4/</code>
<code>-reserve group</code>	<code>data.1/engineering/</code>

- **檔案系統表單** – `fs` 關鍵字可指定保留名稱中的檔案系統元件。[表 3-19](#) 會顯示相關範例。

表 3-19 檔案系統表單範例

指令與關鍵字	預留名稱範例
<code>-reserve fs</code>	<code>proj.1/p103/samfs1</code> <code>proj.1/p104/samfs1</code>

在 `archiver.cmd` 檔案中，您可為一個、兩個或所有三個可能的表單指定 `-reserve` 參數。這三個表單可以組合，並可在歸檔組參數定義中一起使用。

以 [程式碼範例 3-27](#) 中所顯示的 `archiver.cmd` 檔案片段為例，以 `arset.1` 開頭的文字行將會依據歸檔組、群組與檔案系統建立保留的名稱。

程式碼範例 3-27 具有預留磁碟區的 `archiver.cmd` 檔案

```
params
arset.1 -reserve set -reserve group -reserve fs
endparams
```

有關保留磁碟區的資訊將儲存在程式庫目錄中。程式庫目錄中的文字行會列出媒體類型、VSN、保留資訊及保留日期與時間。預留資訊包括由兩個正斜線 (//) 分隔的歸檔組元件、路徑名稱元件及檔案系統元件。

備註 – 這些斜線不代表路徑名稱，而只是在顯示保留名稱之三個元件時所用的分隔符。

如**程式碼範例 3-28** 所示，程式庫目錄中用以說明保留磁碟區的文字行是以 #R 字元為開頭。

程式碼範例 3-28 顯示預留磁碟區的程式庫目錄檔

```
6 00071 00071 lt 0xe8fe 12 9971464 1352412 0x6a000000 131072 0x
# -il-o-b----- 2000/5/24 13:50:02 1969/12/31 18:00:00 2001/7/13 14:03:00
#R lt 00071 arset0.3// 2001/03/19 18:27:31
10 ST0001 NO_BAR_CODE lt 0x2741 9 9968052 8537448 0x68000000 1310
# -il-o----- 07.05.00 15:30:29 1969/12/31 18:00:00 13.04.01 13:46:54
#R lt ST0001 hgm1.1// 2001/03/20 17:53:06
16 SLOT22 NO_BAR_CODE lt 0x76ba 6 9972252 9972252 0x68000000 1310
# -il-o----- 2000/6/6 16:03:05 1969/12/31 18:00:00 2001/7/12 11:02:05
#R lt SLOT22 arset0.2// 2001/03/02 12:11:25
```

備註 – **程式碼範例 3-28** 中的部分文字行已依據頁面大小而予以截斷。

視在 `archiver.cmd` 檔案中定義的選項而定，一個或多個預留資訊欄位可以是空的。日期與時間表示制定預留的時間。於歸檔期間保留給歸檔組之各磁碟區的檔案中，皆會附加保留的文字行。

歸檔器會記錄程式庫目錄檔中的磁碟區預留情況。磁碟區在重新製作標籤時將自動取消預留，因為歸檔資料已被有效刪除。

您也可以使用 `reserve(1M)` 和 `unreserve(1M)` 指令來預留和取消預留磁碟區。如需有關這些指令的更多資訊，請參閱「`reserve(1M)` 線上手冊」和「`unreserve(1M)` 線上手冊」。

您可以使用 `samu(1M)` 公用程式的 `v` 畫面顯示保留資訊，或是使用 `archiver(1M)` 或 `dump_cat(1M)` 指令以**程式碼範例 3-29** 所示的一種格式加以顯示。

程式碼範例 3-29 用於顯示預留資訊的指令

```
archiver -lv
dump_cat -V catalog-name
```

第 100 頁的「範例 4：歸檔至光學媒體的使用者檔案與資料檔案」 顯示使用預留磁碟區的完整歸檔範例。

設定歸檔優先權：-priority

Sun StorageTek SAM 檔案系統提供用於歸檔檔案的可配置優先權系統。每個檔案將指定給可從檔案屬性，及可在 `archiver.cmd` 檔案中為每個歸檔組設定的優先權倍增器計算之優先權。這些屬性包括線上/離線、時間、製作的備份數目及大小。

預設不會對歸檔請求中的檔案進行排序，且所有特性倍增器 (property multiplier) 皆為零。如此檔案將會以先找到者先行歸檔的方式歸檔。您可控制檔案按設定優先權與排序方法歸檔的順序。以下是您可設定的優先權範例：

- 選擇 `priority` 排序方法以便按優先權順序歸檔歸檔請求中的檔案。
- 變更 `archive_loaded` 優先權以減少媒體載入。
- 變更 `offline` 優先權以使線上檔案在離線檔案之前進行歸檔。
- 變更 `copy#` 優先權以便按備份順序製作歸檔備份。

表 3-20 列出歸檔優先權。

表 3-20 歸檔優先權

歸檔優先權	定義
<code>-priority age value</code>	歸檔時間屬性倍增器
<code>-priority archive_immediate value</code>	歸檔立即屬性倍增器
<code>-priority archive_overflow value</code>	多個歸檔磁碟區屬性倍增器
<code>-priority archive_loaded value</code>	歸檔磁碟區載入屬性倍增器
<code>-priority copies value</code>	備份製作屬性倍增器
<code>-priority copy1 value</code>	備份 1 屬性倍增器
<code>-priority copy2 value</code>	備份 2 屬性倍增器
<code>-priority copy3 value</code>	備份 3 屬性倍增器
<code>-priority copy4 value</code>	備份 4 屬性倍增器
<code>-priority offline value</code>	檔案離線屬性倍增器
<code>-priority queuwait value</code>	佇列等待屬性倍增器
<code>-priority rearchive value</code>	重新歸檔屬性倍增器
<code>-priority reqrelease value</code>	Reqrelease 屬性倍增器
<code>-priority size value</code>	檔案大小屬性倍增器
<code>-priority stage_loaded value</code>	分段備份磁碟區載入屬性倍增器
<code>-priority stage_overflow value</code>	多個分段備份磁碟區屬性倍增器

關於 *value*，請在以下範圍內指定浮點數字：

$-3.400000000E+38 \leq value \leq 3.402823466E+38$

如需有關這些優先權的更多資訊，請參閱「archiver(1M) 線上手冊」和「archiver.cmd(4) 線上手冊」。

排定歸檔：-startage、-startcount 與 -startsize

在歸檔器掃描檔案系統時，其會辨識要歸檔的檔案。辨識為歸檔候選者的檔案會列入一個所謂的 *archive request* 清單。在檔案系統掃描結束時，系統會排定歸檔請求以進行歸檔。-startage、-startcount 及 -startsize 歸檔組參數可控制歸檔工作量，以確保可以及時對檔案進行歸檔。表 3-21 會顯示這些參數的格式。

表 3-21 -startage、-startcount 與 -startsize 指令的格式

指令	意義
-startage <i>time</i>	掃描時，第一次將檔案標記成要包含在歸檔請求內，到開始進行歸檔為止所經歷的時間。關於 <i>time</i> ，請以第 65 頁的「設定歸檔時間」中所用的格式指定時間。若是未設定此變數，則會使用 <i>interval</i> 指令。
-startcount <i>count</i>	要包含在歸檔請求內的檔案數量。當歸檔請求中的檔案數量達到 <i>count</i> 的值時，便會開始進行歸檔。依據預設， <i>count</i> 並未設定。
-startsize <i>size</i>	歸檔請求中，要歸檔之所有檔案大小的最小總和（以位元組為單位）。歸檔工作會持續累積，當檔案大小總和達到 <i>size</i> 的值時，便會開始進行歸檔。依據預設， <i>size</i> 並未設定。

examine=method 指令和 *interval=time* 指令可與 -startage、-startcount 和 -startsize 等指令互動。-startage、-startcount 與 -startsize 指令可找出最佳的時機進行歸檔工作。這些值會覆寫 *examine=方法* 規定（如果有的話）。如需有關 *examine* 指令的更多資訊，請參閱第 49 頁的「*examine* 指令：控制歸檔掃描」。如需有關 *interval* 指令的更多資訊，請參閱第 50 頁的「*interval* 指令：指定歸檔間隔」。

archiver.cmd 檔案中只能指定每個歸檔副本的 -startage、-startcount 及 -startsize 指令。您若是指定了其中的多項指令，便會在第一個條件出現時開始進行歸檔作業。若未指定這些指令，便會依據 *examine=method* 指令排定歸檔請求，如下：

- 當 *examine=noscan* 時，預設值為：startage 10 分鐘、startcount 10,000 個、startsize 10 GB。當將第一個檔案存入歸檔請求中之後，即會根據 *interval=time* 指令的指定排定歸檔請求。此為連續歸檔。預設為 *examine=noscan*。
- 當 *examine=scan|scaninodes|scandirs* 時，會將歸檔請求排定在檔案系統掃描後進行歸檔。

「archiver.cmd(4) 線上手冊」含有示範如何使用這些指令的範例。

VSN 關聯指令

archiver.cmd 檔案的 VSN 關聯區段會將磁碟區指定給歸檔組。此區段以 vsns 指令開頭，以 endvsns 指令結尾。

VSN 關聯也可使用 File System Manager 軟體進行配置。如需更多資訊，請參閱 File System Manager 線上說明。

磁碟區集合將被下列表單中的指令指派給歸檔組：

```
archive-set-name .copy-num media-type vsn-expr ... [ -pool vsn-pool-name ... ]
```

表 3-22 VSN 關聯指令的引數

引數	意義
<i>archive-set-name</i>	歸檔組的站點定義名稱。必須是歸檔組指派指令中的第一個欄位。歸檔組名稱通常表示屬於歸檔組的檔案特性。歸檔組名稱限制為字母、數字及底線字元 (_)。不允許使用其他特殊字元或空格。歸檔組名稱的第一個字元必須是字母。
<i>copy-num</i>	其後尾隨一或多個引數指定該備份之歸檔特性的數字。歸檔備份指令以數字開頭。此數字 (1、2、3 或 4) 是備份數目。
<i>media-type</i>	表示媒體類型。如需有效媒體類型清單，請參閱「mcf(4) 線上手冊」。
<i>vsn-expr</i>	表示常規表示式。請參閱「regexp(5) 線上手冊」。
<i>-pool vsn-pool-name</i>	表示已命名的 VSN 集合。

關聯至少需要三個欄位：*archive-set-name* 與 *copy-num*、*media-type*，以及至少一個磁碟區。以小數點號 (.) 連接的 *archive-set-name* 與 *copy-num* 值。

備註 – 如果您的 Sun StorageTek SAM 環境是配置為按歸檔組回收，請勿將 VSN 指定給一個以上的歸檔組。

程式碼範例 3-30 會顯示 VSN 指定的兩個文字行。

程式碼範例 3-30 多行的 VSN 指定

```
vsns
set.1 1t VSN001 VSN002 VSN003 VSN004 VSN005
set.1 1t VSN006 VSN007 VSN008 VSN009 VSN010
endvsns
```

程式碼範例 3-31 會顯示使用反斜線字元 (\) 延續至下一行的 VSN 指定。

程式碼範例 3-31 具有延續行的 VSN 指定

```
vsns
set.1 lt VSN001 VSN002 VSN003 VSN004 VSN005 \
  VSN006 VSN007 VSN008 VSN009 VSN010
endvsns
```

以下範例將使用常規表示式以不同的方法指定相同的 VSN。

程式碼範例 3-32 會以採用速記表示法之常規表示式指定 VSN。

程式碼範例 3-32 使用速記表示法的 VSN 指定

```
vsns
set.1 lt VSN0[1-9] VSN10
endvsns
```

當歸檔器需要磁碟區供歸檔組使用時，會檢查所有自動程式庫與手動掛載磁碟機中之選定媒體類型的各個磁碟區，以判別磁碟區是否符合任何 VSN 表示式的條件。其會選取第一個符合表示式條件，並含有足夠空間可供歸檔備份作業使用的磁碟區。例如：

- 以下指令會指定將隸屬於備份 1 之歸檔組 `ex_set` 的檔案複製到使用 20 個磁碟區中，任一個名稱為 `optic20` 至 `optic39` 之磁碟區的媒體類型 `mo` 上：

```
ex_set.1 mo optic[2-3][0-9]
```

- 以下指令會指定將隸屬於備份 2 的歸檔組 `ex_set` 的檔案複製到使用以 `TAPE` 為開頭之任意磁碟區的媒體類型 `lt` 上：

```
ex_set.2 lt ^TAPE
```

備註 – 在設定 `archiver.cmd` 檔案時，確定已為中介資料將磁碟區指定給歸檔組。每個檔案系統都具有一個與其有相同名稱的歸檔組。如需有關保留中介資料的更多資訊，請參閱「`samfsdump(1M)` 線上手冊」，或參閱「Sun StorageTek Storage Archive Manager 疑難排解指南」。

VSN 池指令

`archiver.cmd` 檔案的 VSN pool 區段，以 `vsnpools` 指令為開頭，並以 `endvsnpools` 指令或 `archiver.cmd` 檔案的尾端做為結尾。此部分可命名磁碟區集合。

VSN 池也可以使用 File System Manager 軟體進行配置。如需更多資訊，請參閱 File System Manager 線上說明。

VSN 池是已命名的磁碟區集合。VSN 池對於定義可用於歸檔組的磁碟區很有用。同樣，VSN 池可提供有用的緩衝區以便將磁碟區指派、預留給歸檔組。您可以使用 VSN 池依據組織內的部門、群組中的使用者、資料類型及其他適用的群組，定義個別的磁碟區群組。

磁碟區已被預留，它將對從其中產生它的池不再可用。因此，已命名池中的磁碟區數目將隨使用的磁碟區變更。您可以採用下列格式輸入 `archiver(1M)` 指令，以檢視 VSN 池：

```
# archiver -lv | more
```

VSN 池定義的語法如下：

```
vsn-pool-name media-type vsn-expr
```

表 3-23 VSN 池指令的引數

引數	意義
<i>vsn-pool-name</i>	VSN 池。
<i>media-type</i>	2 個字元的媒體類型。如需有效媒體類型清單，請參閱「 <code>mc(4)</code> 線上手冊」。
<i>vsn-expr</i>	表示常規表示式。可以有一或多個 <i>vsn-expr</i> 引數。請參閱「 <code>regcmp(3G)</code> 線上手冊」。

以下範例使用四個 VSN 池：users_pool、data_pool、proj_pool 及 scratch_pool。臨時池為一組磁碟區，會在 VSN 關聯中之特定磁碟區用盡，或其他 VSN 池用盡時使用。如果三個特定池的其中之一是來自於磁碟區，歸檔器則會選取 scratch pool VSN。[程式碼範例 3-33](#) 顯示使用四個 VSN 池的 archiver.cmd 檔案。

程式碼範例 3-33 VSN 池

```
vsnpools
users_pool    mo ^MO[0-9][0-9]
data_pool     mo ^DA.*
scratch_pool  mo ^SC[5-9][0-9]
proj_pool     mo ^PR.*
endvsnpools

vsns
users.1       mo    -pool users_pool    -pool scratch_pool
data.1        mo    -pool data_pool     -pool scratch_pool
proj.1        mo    -pool proj_pool     -pool scratch_pool
endvsns
```

如需有關 VSN 關聯的更多資訊，請參閱第 79 頁的「VSN 關聯指令」。

關於磁碟歸檔

歸檔是將檔案從線上磁碟複製到歸檔媒體的程序。進行磁碟歸檔時，歸檔媒體會是檔案系統中的線上磁碟。

可實作磁碟歸檔，以將檔案從某個 Sun StorageTek SAM 檔案系統，歸檔到相同主機電腦上的另一個檔案系統，或是歸檔到另一 Sun Solaris 主機上的其他檔案系統。當使用兩個主機系統實作磁碟歸檔時，這兩個系統會分別擔任何用戶端及伺服器的角色，其中用戶端系統為來源檔的所在，而伺服器系統則是存放歸檔備份的目標系統。

要寫入歸檔檔案的檔案系統可以是任何 UNIX 檔案系統，但若是將磁碟歸檔備份寫入其他主機，便須在該主機上安裝一或多個與 Sun StorageTek SAM 軟體相容的檔案系統。

不論是歸檔至磁碟區的檔案，或是歸檔至程式庫磁碟區的檔案，歸檔器皆會採取相同的處理方式。您仍可製作一、二、三或四個歸檔備份。如果您正在製作多個歸檔備份，其中一個歸檔備份可寫入磁碟區，而其他歸檔備份將寫入可移除的媒體磁碟區。此外，如果您通常歸檔至 Sun StorageTek SAM 檔案系統的磁碟區，歸檔檔案備份將根據此檔案系統中的 archiver.cmd 檔案規則自行歸檔。

下列清單說明歸檔至線上磁碟與歸檔至可移除的媒體之間的某些相似與不同之外：

- 與寫入磁光碟或磁帶的歸檔備份不同，寫入磁碟的歸檔備份不會記錄在目錄中。此外，磁碟區中的歸檔檔案將不出現在記錄器中。
- 如果您正在歸檔至可移除的媒體磁碟區，您可在掛載檔案系統後開始歸檔，而無需變更 `archiver.cmd` 檔案中的任何預設值。反之，您若是歸檔到磁碟區，便須在掛載檔案系統之前，先編輯 `archiver.cmd` 檔案，並定義磁碟歸檔組。
- 磁碟歸檔不會依賴 `mcf` 檔案中的項目。您必須在 `archiver.cmd` 檔案中指定磁碟歸檔組，並在 `/etc/opt/SUNWsamfs/diskvols.conf` 中定義磁碟區。。

備註 – 若只是歸檔到可移除式媒體磁碟區，便無需使用 `diskvols.conf` 配置檔案。

`diskvols.conf` 檔案必須在來源檔案所在的系統中建立。視歸檔備份寫入的位置而定，此檔案亦包含下列資訊：

- 如果歸檔備份被寫入相同主機系統上的檔案系統，`diskvols.conf` 檔案將定義 VSN 與至每個 VSN 的路徑。
- 若是將歸檔備份寫入不同的 Sun Solaris 主機系統，則 `diskvols.conf` 檔案中便會涵括該伺服器系統的主機名稱。在此情況下，伺服器系統上亦必須有 `diskvols.conf` 檔案，伺服器系統可定義被給予寫入此系統權限的用戶端。若要建立此主從關係，請在開始進行第 85 頁的「**啟動磁碟歸檔**」中所述的程序之前，先確定擔任伺服器的主機上，至少已安裝一個 Sun StorageTek SAM 檔案系統。

注意 – 如果您在具有多部 SAM-QFS 伺服器的環境中使用磁碟歸檔，請務必在配置回收器時格外小心。每部 SAM-QFS 伺服器的 `diskvols.conf` 檔案都必須指向一組唯一的磁碟區資源規格 (磁碟歸檔目標目錄)。如果在不同的 SAM-QFS 伺服器之間共用任何規格，則從其中一部 SAM-QFS 伺服器執行回收器會銷毀由其他 SAM-QFS 伺服器所管理的磁碟歸檔資料。

配置設定指引

磁碟歸檔磁碟區的位置雖然沒有限制，但建議將磁碟區設定於原始檔案所在磁碟以外的磁碟上。此外也建議您製作多個歸檔備份，並寫入多種類型的歸檔媒體。例如，您可以將備份 1、備份 2 與備份 3 分別歸檔至磁碟區、磁帶及磁光碟。

若是將檔案歸檔到伺服器系統上的檔案系統，則歸檔檔案本身將可以歸檔到與伺服器相連之程式庫上的可移除式媒體匣中。

磁碟歸檔指令

歸檔到線上磁碟時，歸檔器會識別負責定義歸檔組及配置回收的 `archiver.cmd` 指令。其會忽略可移除式媒體匣的專用指令。具體而言，系統會辨識用於磁碟歸檔組的以下指令：

- 第 66 頁的「歸檔組備份參數」中的所有回收指令，除了以下指令之外：
 - `-fillvsns`
 - `-ovflmin min-size`
 - `-reserve method`
 - `-tapenonstop`
- 第 143 頁的「編輯 `archiver.cmd` 檔案」中的所有指令，除了以下指令之外：
 - `-recycle_dataquantity size`
 - `-recycle_vsncount count`
- `vsns` 和 `endvsns` 指令與 `vsnpools` 和 `endvsnpools` 指令。VSN 關聯章節中支援磁碟區且與 `dk` 媒體檔案定義。磁碟區會以一或多個 VSN 表示式關鍵字表示。您也可以指定 VSN 池，從中選取磁碟區，如程式碼範例 3-34 中所示。

程式碼範例 3-34 `vsns` 與 `vsnpools` 指令範例

```
vsnpools
data_pool dk disk0[0-5]
endvsnpools

vsns
arset0.1 dk disk10 disk1[2-5]
arset1.1 dk -pool data_pool
endvsns
```

從軟體的版本 4U6 開始，磁碟歸檔也可以在 Sun StorageTek 5800 系統上執行。Sun StorageTek 5800 是具有整合式硬體和軟體架構的線上儲存設備，其中的磁碟型儲存節點會編排於對稱叢集中。Sun StorageTek 5800 磁碟歸檔在 `vsns` 指令中的媒體縮寫為 `cb`。

- `clients` 和 `endclients` 指令。您若是將檔案從用戶端主機歸檔到伺服器主機，則伺服器系統上便須具備含有該用戶端系統名稱的 `diskvols.conf` 檔案。程式碼範例 3-35 會顯示這些指令的格式。

程式碼範例 3-35 `clients` 與 `endclients` 指令的格式

```
clients
client-system1
client-system2
...
endclients
```

對於 *client-system*，請指定含有來源檔之用戶端系統的主機名稱。

- `-recycle_minobs percent` 回收器指令。此選項可用於設定磁碟歸檔之回收器重新歸檔程序的臨界值（預設值為 50%）。當磁碟上已歸檔之 `tar` 檔案中過時的檔案的百分比已達此執行緒，回收器會開始從歸檔將有效檔案移動至新的 `tar` 檔案。當所有有效檔案皆經過移動之後，便會將原始的 `tar` 檔案標記為要從磁碟歸檔中移除的候選者。可移除的媒體回收會忽略此選項。
- 進行磁碟歸檔回收時會忽略下列回收指令：
 - `recycle_hwm`
 - `recycle_vsncount`

如需有關磁碟歸檔指令的更多資訊，請參閱「`archiver.cmd(4)` 線上手冊」。

備註 – 如果您使用 Sun StorageTek 5800 上的磁碟區進行歸檔，請注意 Sun StorageTek 5800 不是傳統的檔案系統，因此其安全性考量與其他類型的磁碟儲存裝置不同。請閱讀 Sun StorageTek 5800 文件以取得更多資訊。

▼ 啓動磁碟歸檔

您可以隨時啓動磁碟歸檔。本節中的程序假設您已經在進行歸檔，而且您正將磁碟歸檔新增到您的環境中。如果您將磁碟歸檔啓動做為初始安裝的一部分，相關資訊則請參閱「Sun StorageTek Storage Archive Manager 安裝與升級指南」。

備註 – 在 4U4 版本之前的軟體中，透過 `params` 小節中的 `-disk_archive` 參數可在 `archiver.cmd` 檔案中啓用磁碟歸檔。此參數已不再使用，因此必須編輯舊版軟體所建立的 `archiver.cmd` 檔案，歸檔作業才能夠在版本 4U4 與更新版本中正常運作。請參閱「`archiver.cmd(4)` 線上手冊」以取得詳細資訊。

1. 確定要寫入磁碟歸檔備份的主機上，至少已安裝一個 Sun StorageTek QFS 或 Sun StorageTek SAM 檔案系統。
2. 在含有要歸檔的檔案之主機系統上成為超級使用者。
3. 遵循「Sun StorageTek Storage Archive Manager 安裝與升級指南」中的程序，以在要歸檔之檔案所在的主機上，以及要寫入歸檔備份的主機上使用磁碟歸檔。
4. 在含有要歸檔的檔案之主機上，使用 `samd(1M) config` 指令散佈配置檔案變更，然後重新啟動系統。

```
# samd config
```

5. 若要歸檔到其他主機上的磁碟，請執行下列步驟：
 - a. 在寫入歸檔備份的主機系統上成為超級使用者。
 - b. 使用 `samd(1M) config` 指令傳遞配置檔案的變更，然後重新啟動目標系統。

```
# samd config
```

6. 若要歸檔至 **Sun StorageTek 5800** 系統，您必須升級 **Sun StorageTek 5800** 中介資料模式配置。請遵循「Sun StorageTek 5800 System Administration Guide」中記載的程序進行。

使用[程式碼範例 3-36](#) 中的 XML 重疊，定義 Sun StorageTek SAM 使用的中介資料。

程式碼範例 3-36 STK 5800 上的 Sun StorageTek SAM 中介資料模式

```
<?xml version="1.0" encoding="UTF-8"?>

<metadataConfig>
  <schema>
    <namespace name="com">
      <namespace name="sun">
        <namespace name="samfs">
          <field name="archiveId" type="string" indexable="true"/>
        </namespace>
      </namespace>
    </namespace>
  </schema>

  <fsViews>
</fsViews>

</metadataConfig>
```

磁碟歸檔範例

以下是磁碟歸檔配置的相關範例。

範例 1

在此範例中，標示為 disk01、disk02 與 disk04 的 VSN，會寫入原始來源檔所屬之主機系統 pluto 中。VSN disk03 將被寫入伺服器系統 mars 上的 VSN。

[程式碼範例 3-37](#) 顯示常駐於用戶端系統 pluto 上的 diskvols.conf 檔案。

程式碼範例 3-37 pluto 上的 diskvols.conf 檔案

```
# This is file /etc/opt/SUNWsamfs/diskvols.conf on pluto
# VSN Name      [Host Name:]Path
#
disk01          /sam_arch1
disk02          /sam_arch2/proj_1
disk03          mars:/sam_arch3/proj_3
disk04          /sam_arch4/proj_4
```

[程式碼範例 3-38](#) 顯示伺服器系統 mars 上的 diskvols.conf 檔案。

程式碼範例 3-38 mars 上的 diskvols.conf 檔案

```
# This is file /etc/opt/SUNWsamfs/diskvols.conf on mars
#
clients
pluto
endclients
```

[程式碼範例 3-39](#) 顯示 pluto 上的 archiver.cmd 檔案的一個片段。

程式碼範例 3-39 pluto 上的 archiver.cmd 檔案

```
vsns
arset1.2 dk disk01
arset2.2 dk disk02 disk04
arset3.2 dk disk03
endvsns
```

範例 2

在此範例中，檔案 `/sam1/testdir0/filea` 位於 `arset0.1` 的歸檔組中，而歸檔器會將此檔案的內容複製到目標路徑 `/sam_arch1`。程式碼範例 3-40 會顯示 `diskvols.conf` 檔案。

程式碼範例 3-40 `diskvols.conf` 檔案

```
# This is file /etc/opt/SUNWsamfs/diskvols.conf
#
# VSN Name      [Host Name:]Path
#
disk01          /sam_arch1
disk02          /sam_arch12/proj_1
```

程式碼範例 3-41 顯示針對磁碟歸檔的 `archiver.cmd` 檔案文字行：

程式碼範例 3-41 `archiver.cmd` 檔案中與磁碟歸檔相關的指令

```
.
vsns
arset0.1 dk disk01
endvsns
.
```

程式碼範例 3-42 顯示為歸檔到磁碟之 `filea` 檔案的 `sls(1)` 指令輸出。請注意 `copy 1` 文字行中的下列項目：

- `dk` 為媒體類型或磁碟歸檔媒體
- `disk01` 為 VSN
- `f192` 為磁碟歸檔 `tar(1)` 檔案的路徑

程式碼範例 3-42 `sls(1M)` 的輸出

```
# sls -D /sam1/testdir0/filea
/sam1/testdir0/filea:
mode: -rw-r----- links: 1 owner: root group: other
length: 797904 admin id: 0 inode: 3134.49
archdone;
copy 1: ---- Dec 16 14:03 c0.1354 dk disk01 f192
access: Dec 19 10:29 modification: Dec 16 13:56
changed: Dec 16 13:56 attributes: Dec 19 10:29
creation: Dec 16 13:56 residence: Dec 19 10:32
```


範例 3

在此範例中，檔案 `/sam2/my_proj/fileb` 在歸檔組 `arset0.1` 中的用戶端主機 `snickers` 上，歸檔器會將此檔案內容複製到伺服器主機 `mars` 上的目標路徑 `/sam_arch1`。

程式碼範例 3-43 顯示 `snickers` 上的 `diskvols.conf` 檔案。

程式碼範例 3-43 `snickers` 上的 `diskvols.conf` 檔案

```
# This is file /etc/opt/SUNWsamfs/diskvols.conf on snickers
#
# VSN Name      [Host Name:]Path
#
disk01          mars:/sam_arch1
```

程式碼範例 3-44 顯示 `mars` 上的 `diskvols.conf` 檔案。

程式碼範例 3-44 `mars` 上的 `diskvols.conf` 檔案

```
# This is file /etc/opt/SUNWsamfs/diskvols.conf on mars
#
clients
snickers
endclients
```

程式碼範例 3-45 顯示 `archiver.cmd` 檔案中與此範例相關的指令。

程式碼範例 3-45 `archiver.cmd` 檔案中與磁碟歸檔相關的指令

```
.
vsns
arset0.1 dk disk01
endvsns
.
```

計畫歸檔作業

歸檔器將使用 `archiver.cmd` 檔案自動執行儲存管理作業。寫入此檔案之前，應先查閱適當的一般準則以提升 Sun StorageTek SAM 檔案系統與歸檔器的效能，並進一步確保您的資料會以最安全的方式進行儲存。

- 儲存歸檔記錄。歸檔記錄檔可提供復原資料的必要資訊，甚至在 Sun StorageTek SAM 軟體無法使用時也可提供。請將這些記錄保存在安全的地方，以在 Sun StorageTek SAM 軟體發生災難性損毀而無法使用時使用。
- 為磁碟區使用常規表示式。可讓系統將檔案放置在不同的磁碟區上。磁碟區範圍 (使用常規表示式所指定) 可以讓系統連續執行。對歸檔組備份使用特定的磁碟區名稱，將會快速填滿磁碟區，進而在移除媒體，以更換為其他媒體時發生工作流程過多的問題。
- 您可根據建立與修改檔案的頻率、及是否要儲存所有的修改備份確定歸檔間隔。請記住，歸檔間隔是兩次檔案系統掃描之間的時間。非常短的歸檔間隔可讓歸檔器幾乎保持連續掃描。
- 考量使用的檔案系統數目。一般來說，使用多個 Sun StorageTek SAM 檔案系統時的歸檔器效能，會優於使用單一 Sun StorageTek SAM 檔案系統時的效能。掃描多個檔案系統比掃描單一檔案系統需要的時間相對較少。
- 使用目錄結構組織 Sun StorageTek SAM 檔案系統中的檔案。若要獲致最佳效能，請勿在目錄中存放超過 10,000 個以上的檔案。
- 務必最少在兩個單獨的磁碟區中製作兩個檔案備份。如果媒體發生實體問題，將資料儲存在單一媒體類型上將很危險。
- 確實使用 `samfsdump(1M)` 定期備份您的中介資料 (目錄結構、檔案名稱等)。中介資料會儲存在與檔案系統同名的歸檔組中。您可在發生災難時使用此資訊以復原檔案系統。若不想備份中介資料，可以將此歸檔組指定給不存在的 VSN，以防止對此資料進行歸檔。如需有關預留中介資料的更多資訊，請參閱「Sun StorageTek Storage Archive Manager 疑難排解指南」或「Sun StorageTek Storage Archive Manager 安裝與升級指南」。

預覽佇列

歸檔器與分段備份程序均可請求載入或卸載媒體。如果請求數目超過可用於媒體載入的磁碟機數目，則超過該數目的請求將會傳送至預覽佇列。

預覽佇列中的歸檔與分段備份請求，均為無法立即滿足的請求。依據預設，系統將按先進先出 (FIFO) 順序來滿足預覽請求。

您可以將不同的優先權指派給預覽請求，亦可在預覽程序檔中輸入指令以覆寫 FIFO 預設值。所輸入的指令將寫入 `/etc/opt/SUNWsamfs/preview.cmd`。如需有關此檔案和歸檔與分段備份設定優先權的更多資訊，請參閱第 126 頁的「設定預覽請求的優先權」。

歸檔器範例

本節將提供在實際環境中進行歸檔的相關範例。

範例 1：不具備 archiver.cmd 檔案

此範例將說明歸檔器如何不使用 archiver.cmd 檔案，而在具有一個檔案系統、一個雙磁碟機的光學自動程式庫，以及六個卡匣的 Sun StorageTek SAM 環境中運作。

程式碼範例 3-46 顯示由 archiver(1M) -lv 指令產生的輸出。它顯示歸檔器選定的預設媒體是 mo 類型。只有 mo 媒體可用。

程式碼範例 3-46 顯示歸檔媒體的 archiver(1M) -lv 輸出

```
# archiver -lv
Notify file: /etc/opt/SUNWsamfs/scripts/archiver.sh
Archive media:
media:lt archmax: 512.0M Volume overflow not selected
media:mo archmax: 4.8M Volume overflow not selected
```

程式碼範例 3-47 顯示的輸出表示歸檔器使用兩個磁碟機。它會列出 12 個磁碟區、儲存容量及可用空間。

備註 – archiver(1M) -lv 指令僅於有空間之 VSN 顯示。

程式碼範例 3-47 顯示可用 VSN 的 archiver(1M) -lv 輸出

```
Archive libraries:
Device:hp30 drives_available:2 archive_drives:2
Catalog:
mo.optic00          capacity: 1.2G space: 939.7M -il-o-----
mo.optic01          capacity: 1.2G space: 934.2M -il-o-----
mo.optic02          capacity: 1.2G space: 781.7M -il-o-----
mo.optic03          capacity: 1.2G space: 1.1G -il-o-----
mo.optic10          capacity: 1.2G space: 85.5M -il-o-----
mo.optic11          capacity: 1.2G space: 0 -il-o-----
mo.optic12          capacity: 1.2G space: 618.9k -il-o-----
mo.optic13          capacity: 1.2G space: 981.3M -il-o-----
mo.optic20          capacity: 1.2G space: 1.1G -il-o-----
mo.optic21          capacity: 1.2G space: 1.1G -il-o-----
mo.optic22          capacity: 1.2G space: 244.9k -il-o-----
mo.optic23          capacity: 1.2G space: 1.1G -il-o-----
```

程式碼範例 3-48 會顯示歸檔組 samfs 同時包含了中介資料與資料檔案。在檔案的歸檔時間達到預設的四分鐘 (240 秒) 時，歸檔器將為這些檔案製作一個備份。

程式碼範例 3-48 顯示歸檔檔案選擇的 archiver(1M) -lv 輸出

```
Archive file selections:
Filesystem samfs  Logfile:
samfs Metadata
    copy:1  arch_age:240
samfs1 path:..
    copy:1  arch_age:240
```

程式碼範例 3-49 顯示歸檔組中以指定順序歸檔至磁碟區的檔案。

程式碼範例 3-49 顯示歸檔組與磁碟區的 archiver(1M) -lv 輸出

```
Archive sets:
allsets
samfs.1
media: mo (by default)
Volumes:
    optic00
    optic01
    optic02
    optic03
    optic10
    optic12
    optic13
    optic20
    optic21
    optic22
    optic23
Total space available: 8.1G
```

範例 2：與中介資料分開歸檔的資料檔案

此範例顯示如何將資料檔分成兩個從中介資料中獨立的歸檔組。環境中除了光學自動程式庫之外，還有手動掛載的 DLT 磁帶機。大檔案歸檔至磁帶，而小檔案將歸檔至光碟匣。

程式碼範例 3-50 顯示 archiver.cmd 檔案的內容。

程式碼範例 3-50 顯示 archiver.cmd 檔案的 archiver(1M) -lv 輸出

```
# archiver -lv -c example2.cmd
Reading archiver command file "example2.cmd"
1: # Example 2 archiver command file
2: # Simple selections based on size
3:
4: logfile = /var/opt/SUNWsamfs/archiver/log
5: interval = 5m
6:
7: # File selections.
8: big . -minsize 500k
9: all .
10:    1 30s
11:
12: vsns
13: samfs.1 mo .*0[0-2]          # Metadata to optic00 - optic02
14: all.1 mo .*0[3-9] .*[1-2][0-9] # All others for files
15: big.1 lt .*
16: endvsns
```

程式碼範例 3-51 會顯示所要使用的媒體與磁碟機。

程式碼範例 3-51 顯示媒體和磁碟機的 archiver(1M) -lv 輸出

```
Notify file: /etc/opt/SUNWsamfs/scripts/archiver.sh
Archive media:
media:lt archmax: 512.0M Volume overflow not selected
media:mo archmax: 4.8M Volume overflow not selected
Archive libraries:
Device:hp30 drives_available:0 archive_drives:0
  Catalog:
  mo.optic00      capacity: 1.2G space: 939.7M -il-o-----
  mo.optic01      capacity: 1.2G space: 934.2M -il-o-----
  mo.optic02      capacity: 1.2G space: 781.7M -il-o-----
  mo.optic03      capacity: 1.2G space: 1.1G -il-o-----
  mo.optic04      capacity: 1.2G space: 983.2M -il-o-----
  mo.optic10      capacity: 1.2G space: 85.5M -il-o-----
  mo.optic11      capacity: 1.2G space: 0 -il-o-----
  mo.optic12      capacity: 1.2G space: 618.9k -il-o-----
  mo.optic13      capacity: 1.2G space: 981.3M -il-o-----
  mo.optic20      capacity: 1.2G space: 1.1G -il-o-----
  mo.optic21      capacity: 1.2G space: 1.1G -il-o-----

  mo.optic22      capacity: 1.2G space: 244.9k -il-o-----
  mo.optic23      capacity: 1.2G space: 1.1G -il-o-----
Device:lt40 drives_available:0 archive_drives:0
  Catalog:
  lt.TAPE01      capacity: 9.5G space: 8.5G -il-o-----
  lt.TAPE02      capacity: 9.5G space: 6.2G -il-o-----
  lt.TAPE03      capacity: 9.5G space: 3.6G -il-o-----
  lt.TAPE04      capacity: 9.5G space: 8.5G -il-o-----
  lt.TAPE05      capacity: 9.5G space: 8.5G -il-o-----
  lt.TAPE06      capacity: 9.5G space: 7.4G -il-o-----
```

備註 — archiver(1M) -lv 指令僅於有空間之 VSN 顯示。

程式碼範例 3-52 顯示檔案系統的組織。大於 512000 位元組 (500 KB) 的檔案在四分鐘後歸檔；所有其他檔案在 30 秒後歸檔。

程式碼範例 3-52 顯示檔案系統組織的 archiver(1M) -lv 輸出

```
Archive file selections:
Filesystem samfs  Logfile: /var/opt/SUNWsamfs/archiver/log
samfs Metadata
  copy:1 arch_age:240
big path:. minsize:502.0k
  copy:1 arch_age:240
all path:.
  copy:1 arch_age:30
```

程式碼範例 3-53 會顯示可移除式媒體中的歸檔組切分。

程式碼範例 3-53 顯示歸檔組與可移除式媒體的 `archiver(1M) -lv` 輸出

```
Archive sets:
allsets
all.1
  media: mo
Volumes:
  optic03
  optic04
  optic10
  optic12
  optic13
  optic20
  optic21
  optic22
  optic23
  Total space available:   6.3G
big.1
  media: lt
Volumes:
  TAPE01
  TAPE02
  TAPE03
  TAPE04
  TAPE05
  TAPE06
  Total space available:  42.8G
samfs.1
  media: mo
Volumes:
  optic00
  optic01
  optic02
  Total space available:   2.6G
```

範例 3：歸檔至各式媒體的使用者檔案與資料檔案

在此範例中，使用者檔案與專案資料檔將歸檔至各種媒體。data 目錄中的檔案將被按大小與光學及磁帶媒體隔離。指定給群組 ID pict 的檔案將被指定給其他組的磁碟區。tmp 與 users/bob 目錄中的檔案將不歸檔。歸檔將以 15 分鐘的間隔執行，並保留歸檔記錄。

程式碼範例 3-54 會顯示此範例中之 archiver(1M) -lv -c 指令的輸出。

程式碼範例 3-54 archiver(1M) -lv -c 指令輸出

```
# archiver -lv -c example3.cmd
Reading archiver command file "example3.cmd"
1: # Example 3 archiver command file
2: # Segregation of users and data
3:
4: interval = 30s
5: logfile = /var/opt/SUNWsamfs/archiver/log
6:
7: no_archive tmp
8:
9: fs = samfs
10: no_archive users/bob
11: prod_big data -minsize 50k
12:   1 1m 30d
13:   2 3m
14: prod data
15:   1 1m
16: proj_1 projs/proj_1
17:   1 1m
18:   2 1m
19: joe . -user joe
20:   1 1m
21:   2 1m
22: pict . -group pict
23:   1 1m
24:   2 1m
25:
26: params
27: prod_big.1 -drives 2
28: prod_big.2 -drives 2
29: endparams
30:
31: vsns
32: samfs.1 mo optic0[0-1]$
```


程式碼範例 3-54 archiver(1M) -lv -c 指令輸出 (續)

```
33: joe.1 mo optic01$
34: pict.1 mo optic02$
35: pict.2 mo optic03$
36: proj_1.1 mo optic1[0-1]$
37: proj_1.2 mo optic1[2-3]$
38: prod.1 mo optic2.$
39: joe.2 lt 0[1-2]$
40: prod_big.1 lt 0[3-4]$
41: prod_big.2 lt 0[5-6]$
42: endvsns
Notify file: /etc/opt/SUNWsamfs/scripts/archiver.sh
Archive media:
media:lt archmax: 512.0M Volume overflow not selected
media:mo archmax: 4.8M Volume overflow not selected
Archive libraries:
Device:hp30 drives_available:0 archive_drives:0
Catalog:
  mo.optic00      capacity: 1.2G space: 939.7M -il-o-----
  mo.optic01      capacity: 1.2G space: 934.2M -il-o-----
  mo.optic02      capacity: 1.2G space: 781.7M -il-o-----
  mo.optic03      capacity: 1.2G space: 1.1G -il-o-----
  mo.optic04      capacity: 1.2G space: 983.2M -il-o-----
  mo.optic10      capacity: 1.2G space: 85.5M -il-o-----
  mo.optic11      capacity: 1.2G space: 0 -il-o-----
  mo.optic12      capacity: 1.2G space: 618.9k -il-o-----
  mo.optic13      capacity: 1.2G space: 981.3M -il-o-----
  mo.optic20      capacity: 1.2G space: 1.1G -il-o-----
  mo.optic21      capacity: 1.2G space: 1.1G -il-o-----
  mo.optic22      capacity: 1.2G space: 244.9k -il-o-----
  mo.optic23      capacity: 1.2G space: 1.1G -il-o-----
Device:lt40 drives_available:0 archive_drives:0
Catalog:
  lt.TAPE01      capacity: 9.5G space: 8.5G -il-o-----
  lt.TAPE02      capacity: 9.5G space: 6.2G -il-o-----
  lt.TAPE03      capacity: 9.5G space: 3.6G -il-o-----
  lt.TAPE04      capacity: 9.5G space: 8.5G -il-o-----
  lt.TAPE05      capacity: 9.5G space: 8.5G -il-o-----
  lt.TAPE06      capacity: 9.5G space: 7.4G -il-o-----
Archive file selections:
Filesystem samfs Logfile: /var/opt/SUNWsamfs/archiver/log
samfs Metadata
  copy:1 arch_age:240
no_archive Noarchive path:users/bob
prod_big path:data minsize:50.2k
  copy:1 arch_age:60 unarch_age:2592000
  copy:2 arch_age:180
```

程式碼範例 3-54 archiver(1M) -lv -c 指令輸出 (續)

```
prod path:data
  copy:1 arch_age:60
proj_1 path:projs/proj_1
  copy:1 arch_age:60
  copy:2 arch_age:60
joe path:.. uid:10006
  copy:1 arch_age:60
  copy:2 arch_age:60
pict path:.. gid:8005
  copy:1 arch_age:60
  copy:2 arch_age:60
no_archive Noarchive path:tmp
samfs path:..
  copy:1 arch_age:240
Archive sets:
allsets
joe.1
  media: mo
  Volumes:
    optic01
  Total space available: 934.2M
joe.2
  media: lt
  Volumes:
    TAPE01
    TAPE02
  Total space available: 14.7G
pict.1
  media: mo
  Volumes:
    optic02
  Total space available: 781.7M
pict.2
  media: mo
  Volumes:
    optic03
  Total space available: 1.1G
prod.1
  media: mo
  Volumes:
    optic20
    optic21
    optic22
    optic23
  Total space available: 3.3G
```

程式碼範例 3-54 archiver(1M) -lv -c 指令輸出 (續)

```
prod_big.1
media:lt drives:2
Volumes:
  TAPE03
  TAPE04
Total space available: 12.1G
prod_big.2
media: lt drives:2
Volumes:
  TAPE05
  TAPE06
Total space available: 16.0G
proj_1.1
media: mo
Volumes:
  optic10
Total space available: 85.5M
proj_1.2
media: mo
Volumes:
  optic12
  optic13
Total space available: 981.9M
samfs.1
media: mo
Volumes:
  optic00
  optic01
Total space available: 1.8G
```

範例 4：歸檔至光學媒體的使用者檔案與資料檔案

在此範例中，使用者檔案與專案資料檔將歸檔至光學媒體。

將定義四個 VSN 池；三個池用於使用者、資料及專案，一個是 `scratch pool`。當 `proj_pool` 的媒體用盡時，會使用 `scratch_pool` 保留磁碟區。此範例顯示如何基於歸檔組元件、擁有者元件及檔案系統元件為每個歸檔組預留磁碟區。歸檔將以 10 分鐘的間隔執行，並保留歸檔記錄。

[程式碼範例 3-55](#) 顯示 `archiver.cmd` 檔案和歸檔器輸出。

程式碼範例 3-55 `archiver.cmd` 檔案和歸檔器輸出

```
Reading archiver command file "example4.cmd"
1: # Example 4 archiver command file
2: # Using 4 VSN pools
3:
4: interval = 30s
5: logfile = /var/opt/SUNWsamfs/archiver/log
6:
7: fs = samfs
8: users users
9:     1 10m
10:
11: data data
12:     1 10m
13:
14: proj projects
15:     1 10m
16:
17: params
18: users.1 -reserve user
19: data.1 -reserve group
20: proj.1 -reserve dir -reserve fs
21: endparams
22:
23: vsnpools
24: users_pool mo optic0[1-3]$
25: data_pool mo optic1[0-1]$
26: proj_pool mo optic1[2-3]$
27: scratch_pool mo optic2.$
28: endvsnpools
29:
30: vsn
31: samfs.1 mo optic00
32: users.1 mo -pool users_pool -pool scratch_pool
33: data.1 mo -pool data_pool -pool scratch_pool
34: proj.1 mo -pool proj_pool -pool scratch_pool
35: endvsns
```

程式碼範例 3-55 archiver.cmd 檔案和歸檔器輸出 (續)

```
Notify file: /etc/opt/SUNWsamfs/scripts/archiver.sh
Archive media:
media:mo archmax: 4.8M Volume overflow not selected
Archive libraries:
Device:hp30 drives_available:0 archive_drives:0
Catalog:
mo.optic00 capacity: 1.2G space: 939.7M -il-o-----
mo.optic01 capacity: 1.2G space: 934.2M -il-o-----
mo.optic02 capacity: 1.2G space: 781.7M -il-o-----
mo.optic03 capacity: 1.2G space: 1.1G -il-o-----
mo.optic04 capacity: 1.2G space: 983.2M -il-o-----
mo.optic10 capacity: 1.2G space: 85.5M -il-o-----
mo.optic11 capacity: 1.2G space: 0 -il-o-----
mo.optic12 capacity: 1.2G space: 618.9k -il-o-----
mo.optic13 capacity: 1.2G space: 981.3M -il-o-----
mo.optic20 capacity: 1.2G space: 1.1G -il-o-----
mo.optic21 capacity: 1.2G space: 1.1G -il-o-----
mo.optic22 capacity: 1.2G space: 244.9k -il-o-----
mo.optic23 capacity: 1.2G space: 1.1G -il-o-----
Archive file selections:
Filesystem samfs Logfile: /var/opt/SUNWsamfs/archiver/log
samfs Metadata
copy:1 arch_age:240
users path:users
copy:1 arch_age:600
data path:data
copy:1 arch_age:600
proj path:projects
copy:1 arch_age:600
samfs path:.
copy:1 arch_age:240
VSN pools:
data_pool media: mo Volumes:
optic10
Total space available: 85.5M
proj_pool media: mo Volumes:
optic12
optic13
Total space available: 981.9M
scratch_pool media: mo Volumes:
optic20
optic21
optic22
optic23
Total space available: 3.3G
```

程式碼範例 3-55 archiver.cmd 檔案和歸檔器輸出 (續)

```
users_pool media: mo Volumes:
  optic01
  optic02
  optic03
  Total space available: 2.7G
Archive sets:
allsets
data.1
  reserve:/group/
  media: mo
  Volumes:
    optic10
    optic20
    optic21
    optic22
    optic23
  Total space available: 3.4G
proj.1
  reserve:/dir/fs
  media: mo
  Volumes:
    optic12
    optic13
    optic20
    optic21
    optic22
    optic23
  Total space available: 4.2G
samfs.1
  media: mo
  Volumes:
    optic00
  Total space available: 939.7M
users.1
  reserve:/user/
  media: mo
  Volumes:
    optic01
    optic02
    optic03
    optic20
    optic21
    optic22
    optic23
  Total space available: 6.0G
```

第 4 章

釋放

釋放是指釋放器藉由識別已歸檔的檔案並釋放其磁碟快取記憶體備份，以取得可用磁碟快取記憶體空間的程序。這樣即可騰出空間，用於從歸檔媒體建立或分段備份其他檔案。釋放器僅可以釋放歸檔檔案。已釋放的檔案在磁碟快取記憶體中沒有資料。

當達到站點所指定的磁碟臨界值時，Sun StorageTek SAM 軟體即會自動呼叫釋放器程序。或者，您可以使用 `release(1)` 指令立即釋放檔案的磁碟空間，或設定檔案的釋放參數。

您也可以指定在歸檔後立即釋放檔案、永不釋放檔案或只釋放部分檔案。

本章將說明釋放程序與釋放器作業。本附錄包含以下各節：

- [第 104 頁的「釋放器程序簡介」](#)
- [第 105 頁的「釋放器概念」](#)
- [第 106 頁的「關於部分釋放和部分分段備份」](#)
- [第 109 頁的「關於 `releaser.cmd` 檔案」](#)
- [第 116 頁的「規劃釋放器作業」](#)
- [第 117 頁的「手動執行釋放器」](#)

釋放器程序簡介

當檔案系統使用率超出其所配置的高標記時，檔案系統管理軟體便會呼叫釋放器執行下列操作：

1. 讀取 `releaser.cmd` 檔案並收集負責控制釋放程序的指令。
2. 掃描檔案系統並收集各檔案的相關資訊。
3. 依據優先權順序開始釋放檔案。

一個檔案系統可以包含數千個檔案。追蹤各檔案的釋放優先權可能不具效率，因為光是釋放幾個大型檔案，即可能造成檔案系統回到低標記。但是，釋放器必須檢查每個檔案的優先權，否則可能會遺漏最佳釋放候選者。為此，釋放器會先識別前 10,000 個候選者。後續候選者的優先權只要低於前 10,000 個候選者中的最低優先權者，即會遭到捨棄。

釋放器在確定前 10,000 候選者的優先權後，會選擇優先權最高的檔案進行釋放。釋放各檔案之後，釋放器會檢查檔案系統快取記憶體使用率是否低於低標記。若是低於低標記，釋放器便會停止釋放檔案。若是高於低標記，釋放器便會依據優先權順序繼續釋放檔案。

如果在釋放器釋放所有 10,000 個候選者後，檔案系統仍高於低標記，釋放器會重新開始並再找 10,000 個新的候選者。

如果找不到任何可用的候選者，釋放器就會結束。這種情況有可能會發生，例如在檔案沒有歸檔備份時。若是發生，Sun StorageTek SAM 軟體會在一分鐘過後開始再次釋放。

高標記和低標記會以 `high=percent` 和 `low=percent` 檔案系統掛載選項進行設定。如需有關掛載選項的更多資訊，請參閱「`mount_samfs(1M)` 線上手冊」。

釋放器概念

本節將說明釋放器程序的基本概念：

- 時間 — 從事件發生到目前為止所經過的時間。檔案的 `inode` 會持續追蹤釋放器所使用的以下各種時間：
 - 常駐變更時間
 - 資料修改時間
 - 資料存取時間

您可以使用 `s1s(1)` 指令的 `-D` 選項來檢視這些時間。每個時間均有相應的經過時間。例如：如果現在是 10:15 a.m.，檔案的修改時間是 10:10 a.m.，則資料修改時間是 5 分鐘。如需有關 `s1s(1)` 指令的更多資訊，請參閱「`s1s(1)` 線上手冊」。

- 候選者 — 符合釋放條件的候選檔案。在下列情況下，檔案不會成爲候選者：
 - 檔案已離線。
 - 檔案尚未歸檔。
 - `archiver.cmd` 指令檔案爲該檔案指定了 `-norelease` 屬性，且尚未製作要求的備份。
 - 檔案已標記爲「損壞」。
 - 檔案不是一般檔案，而是目錄、區段、特殊字元檔案或管線。
 - 歸檔器正在分段備份檔案以製作額外的備份。在歸檔器分段備份檔案之後，該檔案即可釋放。
 - 檔案時間爲負數。時鐘設定不正確的網路檔案系統 (NFS) 用戶端可能會發生此情況。
 - 檔案被標記爲永不釋放。您可以使用 `release(1) -n` 指令來進行指定。
 - 檔案在過去的某個時間進行分段備份，且該時間短於最短常駐時間設定。如需更多資訊，請參閱第 113 頁的「`min_residence_age` 指令：指定最短常駐時間」。
 - 檔案已透過 `release(1)` 指令的 `-p` 選項標記爲部分釋放，並且已完成部分釋放。
 - 檔案太小。將其釋放不會產生大量空間。
- 優先權 — 優先權爲數值，可依據使用者所提供要套用至該候選者之數值屬性上的權數，指定候選檔案的等級。總優先權爲以下兩種優先權的總合：時間優先權和大小優先權。

系統將先釋放優先權數值較大的候選檔案，然後再釋放優先權數值較小的候選檔案。

- 權數 — 權數爲數值，可偏壓優先權計算，以併入和您相關的檔案屬性並排除與您無關的檔案屬性。例如，若將大小權數設定爲 0，便會將檔案的大小屬性排除在優先權計算之外。權數爲 0.0 到 1.0 之間的浮點值。
- 部分釋放 — 使用部分釋放會將檔案的開頭部分保留在磁碟快取記憶體中，而將其餘部分進行釋放。部分釋放可搭配只讀取檔案開頭的公用程式 (如 `filemgr(1)`) 一起使用。

關於部分釋放和部分分段備份

釋放與分段備份均為補充程序。檔案在歸檔後可以從線上磁碟快取記憶體中完全釋放；站點亦可指定將檔案的開頭部分 (*stub*) 留在磁碟快取記憶體中，而釋放其餘部分。部分釋放檔案可不分段備份檔案，而直接存取檔案 *stub* 中的資料。

您若為系統管理員，即可在檔案系統掛載時，指定將 *stub* 的預設部分釋放大小與最大大小保持線上狀態。您可以透過 `mount(1M)` 指令，或在 `File System Manager` 軟體中設定這些值。如需更多資訊，請參閱 `File System Manager` 線上說明。

`mount(1M)` 指令的選項如下：

- `-o partial=n` 選項 — 設定要保持線上狀態的檔案 *stub* 預設大小 (*n*)。
`-o partial=n` 設定必須小於或等於 `-o maxpartial=n` 設定。
- `-o maxpartial=n` 選項 — 設定要保持線上狀態的檔案 *stub* 最大大小 (*n*)。

您可以在 `release(1)` 指令中指定 `-p` 選項，或在 `sam_release(3)` 程式庫常式中指定 `p` 選項，以指定檔案的預設 *stub* 大小。要為不同類型的檔案或不同應用程式指定不同大小的檔案 *stub*，使用者可以在 `release(1)` 指令中指定 `-s` 選項或在 `sam_release(3)` 程式庫常式中指定 `s` 選項。`-s` 與 `s` 值必須小於掛載檔案系統時，`mount(1M)` 指令中所使用的 `-o maxpartial` 值。

備註 — 即使部分釋放檔案，此檔案仍會在磁碟上佔用大小等於一個 DAU 的空間。例如，如果將部分釋放檔案 *stub* 設定為 16 K 而 DAU 大小為 256 K，則即使已部分釋放檔案，此檔案在磁碟上仍將佔用 256 K 的實際空間。

另一個掛載選項 `-o partial_stage=n` 可讓系統管理員建立在分段備份檔案其餘部分之前，必須讀取的部分釋放 *stub* 數目。讀取超過 `-o partial_stage=n` 大小的指定，將會初始化檔案分段備份。

依據預設，`-o partial_stage=n` 選項將設定為部分釋放 *stub* 的大小。變更此值會影響檔案分段備份，如下所示：

- 如果 `-o partial_stage=n` 選項設定為部分釋放 *stub* 的大小，在應用程式達到部分釋放 *stub* 的結尾之前，預設運作方式會防止分段備份檔案。在達到 *stub* 結尾之前持續等待，將會導致在存取檔案其餘部分時出現延遲。
- 若 `-o partial_stage=n` 選項的設定小於部分釋放 *stub* 的值，便會在應用程式超過 `-o partial_stage=n` 選項所設定的臨界值時執行檔案分段備份。這樣就降低了在存取檔案資料其餘部分時發生延遲的可能性。

例如，假設以下是有效的選項：

- `-o partial_stage=16` (16 KB)
- `-o partial=2097152` (2 GB)
- `-o maxpartial=2097152` (2 GB)

`filemgr(1)` 程式會讀取檔案的前 8 KB。該檔案沒有分段備份。即時影像播放程式讀取相同的檔案，在讀取檔案的前 16 KB 之後，該檔案被分段備份。在歸檔磁帶掛載並定位時，該應用程式將繼續讀取 2 GB 的磁碟資料。在即時影像播放程式讀取 2 GB 的檔案資料之後，它會在完成分段備份作業後立即進行讀取。應用程式無須等待，因為磁帶掛載與定位工作在應用程式讀取部分檔案資料時即已執行。

有多個命令行選項可影響檔案能否標記為部分釋放。某些選項由系統管理員啟動，其他選項則可由個別使用者啟動。下列章節說明可由各種使用者設定的釋放特性。

系統管理員選項摘要

您若為系統管理員，可以在掛載檔案系統時，變更部分釋放的最大值與預設值。[表 4-1](#) 中的 `mount(1M)` 選項會影響部分釋放。如需有關 `mount(1)` 指令的更多資訊，請參閱「`mount_samfs(1M)` 線上手冊」。

表 4-1 部分釋放的掛載選項

<code>mount(1M)</code> 選項	作用
<code>-o maxpartial=n</code>	<p>在檔案標記為部分釋放時，決定在線上磁碟快取記憶體中可保留的最大空間容量 (KB)。最大值是 2,097,152 KB，即 2 GB。最小值是 0，它會防止部分釋放任何檔案。</p> <p>如果指定 <code>-o maxpartial=0</code>，部分釋放功能將停用，被釋放的檔案將會完全釋放，該檔案的任何部分均不會保留在磁碟快取記憶體中。在檔案系統掛載後，使用者將無法覆寫在此選項中指定的值。</p> <p>依據預設，<code>n</code> 引數會設為 16。此設定可讓使用者以磁碟上保留的最大數量 (16 KB) 來標記要部分釋放的檔案。</p>
<code>-o partial=n</code>	<p>設定在使用者使用 <code>release(1)</code> 指令的 <code>-p</code> 選項將檔案標記為部分釋放時，在磁碟快取記憶體中保留的預設空間容量 (KB)。<code>n</code> 引數必須至少為 8，最大可等於為 <code>-o maxpartial=n</code> 選項指定的值。</p> <p>由於某些應用程式無需存取整個檔案即可完成工作，因此該選項可用於確保應用程式具有必需的檔案開頭部分。此外，使用此選項亦可防止對檔案進行不必要的分段備份。</p> <p>依據預設，<code>n</code> 為 16。</p> <p>備註：即使已從磁碟部分釋放了檔案，此檔案仍會在磁碟上佔用大小等於一個 DAU 的空間。</p>
<code>-o partial_stage=n</code>	<p>指定在存取部分釋放的檔案時，必須先讀取 <code>n</code> 位元組的檔案，然後再從歸檔媒體中分段備份整個檔案。此值通常設定為低於 <code>-o partial</code> 設定的容量。對於 <code>n</code>，請指定 0 至 <code>-o maxpartial</code> 指定值之間的整數。預設此值為 16，或任何針對 <code>-o partial</code> 選項而指定的值。</p>
<code>-o stage_n_window=n</code>	<p>將在某段任意時間內要分段備份的資料容量指定為 <code>n</code>。關於 <code>n</code>，請指定從 64 至 2,048,000 的整數值。預設值為 256 KB。此選項僅適用於已設定 <code>stage -n</code> 屬性的檔案。</p>

使用者選項摘要

您若為系統管理員，可以設定在釋放檔案之後，要保留在磁碟快取記憶體中之檔案 **stub** 大小的最大值與預設值。您也可以決定是否要啓用特定檔案系統的部分釋放功能。

但是，使用者可以透過使用 `release(1)` 指令與 `sam_release(3)` 程式庫常式，來設定其他釋放屬性，並可指定要標記為部分釋放的檔案。表 4-2 中會顯示用以決定部分釋放屬性的指令與程式庫選項。如需有關 `release(1)` 指令的更多資訊，請參閱「`release(1)` 線上手冊」。如需有關 `sam_release(3)` 程式庫常式的更多資訊，請參閱「`sam_release(3)` 線上手冊」。

表 4-2 使用者釋放選項

選項	作用
<code>release(1)</code> 指令和 <code>-p</code> 選項 或 <code>sam_release(3)</code> 程式庫常式和 <code>p</code> 選項	<code>-p</code> 與 <code>p</code> 選項可將指定的檔案標記為部分釋放。如果使用了這兩個選項，檔案被釋放後保留在線上磁碟快取記憶體中的檔案容量，將取決於在掛載該檔案所在的檔案系統時設定的 <code>-o partial=n</code> 選項值。這兩個選項不能用於指定保持線上狀態的位元組數目。
<code>release(1)</code> 指令和 <code>-s partial_size</code> 選項 或 <code>sam_release(3)</code> 程式庫常式和 <code>s</code> 選項	<code>-s</code> 與 <code>s</code> 選項可將指定的檔案標記為部分釋放，並可指定保留在線上磁碟快取記憶體中的檔案容量。 <code>-s</code> 或 <code>s</code> 選項的引數將指定保持線上狀態的容量 (KB)。使用者為保持線上狀態的檔案指定的容量，不能大於在掛載檔案系統時為 <code>-o maxpartial=n</code> 值指定的容量。如果使用者指定的值大於檔案系統的值，系統將使用檔案系統的值，而忽略使用者指定的值。

關於 `releaser.cmd` 檔案

`/etc/opt/SUNWsamfs/releaser.cmd` 檔案是由負責指定站點特定之釋放動作的指令行所組成。`releaser.cmd` 檔案可以包含設定釋放優先權、指定記錄檔及其他動作的指令。

下列幾節將說明 `releaser.cmd` 指令：

- 第 109 頁的「指定與時間和大小相關的釋放優先權指令」
- 第 112 頁的「`fs` 指令：指定個別檔案系統的指令」
- 第 112 頁的「`no_release` 和 `display_all_candidates` 指令：指定除錯」
- 第 113 頁的「`min_residence_age` 指令：指定最短常駐時間」
- 第 113 頁的「`logfile` 指令：指定記錄檔」
- 第 114 頁的「`rearch_no_release` 指令：限制釋放重新歸檔的檔案」
- 第 115 頁的「`list_size` 指令：調整釋放器候選者清單的大小」
- 第 115 頁的「指定歸檔組中所有檔案的釋放屬性」

如需有關這些指令的更多資訊，請參閱「`releaser.cmd(4)` 線上手冊」。

透過使用 File System Manager 軟體，可對部分全域釋放指令進行配置。如需更多資訊，請參閱 File System Manager 線上說明。

指定與時間和大小相關的釋放優先權指令

釋放器會依據 `releaser.cmd` 檔案中所定義之指令決定的優先權順序從檔案系統中釋放檔案。在此過程中，系統將同時考量檔案時間與大小。依據預設，站台將先釋放最大且最早的檔案，而將最小且最新的檔案留在磁碟上。下列章節說明釋放器在決定檔案系統中的檔案釋放優先權時，如何考量檔案的時間與大小。

若需更多有關釋放器指令的資訊，請參閱「`releaser.cmd(4)` 線上手冊」。

檔案時間

在決定檔案釋放優先權中與時間相關的元件時，釋放器將考量下列可能的時間：

- 自上次存取之後的時間
- 自上次修改之後的時間
- 在磁碟快取記憶體中變更常駐之後的時間

預設的檔案時間是這三個檔案時間中較晚的一個。

在某些情況下，僅會取用從最近的存取時間、修改時間與常駐變更時間衍生而來的時間。在其他情況下，您可能會優先考量檔案的存取時間，而非修改時間。您可以使用指令來指定在計算檔案的釋放優先權時，使用加權後的時間優先權。

程式碼範例 4-1 會顯示時間優先權指令的格式。

程式碼範例 4-1 時間優先權指令格式

```
weight_age = float
weight_age_access = float
weight_age_modification = float
weight_age_residence = float
```

- `weight_age` 指令可指定針對檔案的預設時間 (檔案存取時間、修改時間或常駐時間中的較小值) 給定加權係數。對於 `float`，請指定 0.0 到 1.0 之間的浮點數字。依據預設，`float = 1.0`。

此指令不可與 `weight_age_access`、`weight_age_modification` 或 `weight_age_residence` 指令一起指定。

- `weight_age_access`、`weight_age_modification` 與 `weight_age_residence` 指令可指定將檔案的時間交由這三個可能時間的一個、兩個或三個決定。對於 `float`，請指定 0.0 到 1.0 之間的浮點數字。依據預設，`float = 1.0`。

這些指令不能與 `weight_age` 指令一起指定。

若同時使用 `weight_age_access`、`weight_age_modification` 與 `weight_age_residence` 指令，便會以下列方式計算檔案時間相關的優先權：

1. 收集每一個檔案之可能時間的檔案時間資料。
2. 將檔案時間資料與 `releaser.cmd` 檔案中所指定的加權係數相乘。
3. 將相乘的乘積加總，如下列方程式所示。

```
file access age * weight_age_access
+ file modification age * weight_age_modification
+ file residency age * weight_age_residence
-----
= age_related_priority
```

程式碼範例 4-2 會顯示 `releaser.cmd` 檔案中的文字行，這些文字行會指定在計算檔案的釋放優先權時，只考量檔案的常駐時間，而不考慮修改時間與存取時間。

程式碼範例 4-2 `releaser.cmd` 檔案片段

```
weight_age_residence = 1.0
weight_age_modify = 0.0
weight_age_access = 0.0
```

計算出檔案與時間相關的優先權後，再將其乘以檔案與大小相關的優先權。與大小相關的優先權會以下節說明的方式計算。

檔案大小

在決定檔案釋放優先權中與大小相關的元件時，釋放器將考量檔案的大小。將檔案大小(以 4 KB 區段為單位) 乘以針對 `weight_size` 指令所指定的權數，以決定檔案釋放優先權中與大小相關的元件。

`weight_size` 指令的格式如下：

```
weight_size = float
```

對於 *float*，請指定 0.0 到 1.0 之間的浮點數字。依據預設 *float* = 1.0。

程式碼範例 4-3 會顯示 `releaser.cmd` 檔案，該檔案會指定在計算檔案的釋放優先權時，忽略 `samfs1` 與 `samfs2` 檔案系統中之所有檔案的檔案大小。

程式碼範例 4-3 `releaser.cmd` 檔案

```
# releaser.cmd file
logfile = /var/adm/default.releaser.log
weight_size = 0.0
#
fs = samfs1
weight_age = 1.0
logfile = /var/adm/samfs1.releaser.log
#
fs = samfs2
weight_age_modify = 0.3
weight_age_access = 0.03
weight_age_residence = 1.0
logfile = /var/adm/samfs2.releaser.log
```

fs 指令：指定個別檔案系統的指令

您可以在 `releaser.cmd` 檔案中使用 `fs = family-set-name` 指令，以表示 `fs =` 指令之後的指令僅適用於指定的檔案系統。

此指令具有以下格式：

```
fs = family-set-name
```

對於 `family-set-name`，請指定 `mcf` 檔案中的系列組名稱。

第一個 `fs =` 指令之前的所有指令均為全域指令，適用於所有檔案系統。`fs =` 指令之後的所有指令將覆寫全域指令。本章所述的指令可用做全域指令或特定於某一檔案系統的指令。

「`releaser.cmd(4)` 線上手冊」包含 `fs =` 指令的範例。

no_release 和 display_all_candidates 指令：指定除錯

在對釋放器進行調校或除錯時，可使用 `no_release` 與 `display_all_candidates` 指令。這些指令如下所示：

- `no_release` 指令可防止從線上磁碟快取記憶體中移除檔案。您可以在不釋放檔案的情況下，使用此指令來檢查 `releaser.cmd` 中的指令。

此指令具有以下格式：

```
no_release
```

- `display_all_candidates` 指令可將所有釋放候選者的名稱寫入日誌檔。

此指令具有以下格式：

```
display_all_candidates
```

這些指令在除錯時非常有用，因為釋放器只會將釋放候選者的名稱寫入記錄檔，而不會實際將其從檔案系統中釋放。

min_residence_age 指令：指定最短常駐時間

`min_residence_age` 指令可讓您指定檔案在成為釋放候選者之前，必須常駐於檔案系統內的最短時間。

此指令具有以下格式：

```
min_residence_age = time
```

關於 *time*，請指定以秒為單位的時間。預設時間是 600，即 10 分鐘。沒有實用的最短或最長 *time* 設定。

logfile 指令：指定記錄檔

若在 `releaser.cmd` 檔案中指定了 `logfile` 指令，釋放器會將其作業記錄附加至指定的檔案中；若指定的檔案不存在，即會建立該檔案。

此指令具有以下格式：

```
logfile = filename
```

關於 *filename*，請指定記錄檔的名稱。

[程式碼範例 4-4](#) 顯示的是記錄檔範例。

程式碼範例 4-4 釋放器記錄檔範例

```
Releaser begins at Wed Apr 28 17:29:06 2006
inode pathname          /sam1/.inodes
low-water mark          24%
weight_size             1
weight_age              1
fs equipment ordinal    1
family-set name         samfs1
started by sam-amld?    yes
release files?          yes
display_all_candidates? no
---before scan---
blocks_now_free:        3481504
lwm_blocks:             3729362
---scanning---
10501 (R:Wed Apr 21 18:47:50 CDT 2006) 10001 min, 500 blks /sam1/testdir0/filevp
10500 (R:Wed Apr 21 18:48:10 CDT 2006) 10000 min, 500 blks /sam1/testdir0/filewq
```

```
...
---after scan---
blocks_now_free:      3730736
lwm_blocks:          3729362
archnodrop: 0
already_offline: 0
bad_inode_number: 0
damaged: 0
extension_inode: 0
negative_age: 0
nodrop: 1
not_regular: 9
number_in_list: 675
released_files: 202
too_new_residence_time: 0
too_small: 2
total_candidates: 675
total_inodes: 1376
wrong_inode_number: 0
zero_arch_status: 689
zero_inode_number: 0
zero_mode: 0
CPU time: 2 seconds.
Elapsed time: 10 seconds.
Releaser ends at Wed Apr 28 17:29:16 2006
```

「releaser(1M) 線上手冊」說明了記錄檔中包含的資訊。由於記錄大小會隨著釋放器每一次的執行而增加，因此您應規劃記錄檔的重建，或省略 logfile 關鍵字。

research_no_release 指令：限制釋放重新歸檔的檔案

依據預設，系統將釋放標記為重新歸檔的檔案。如果在 releaser.cmd 檔案中指定了 research_no_release 指令，則釋放器不會釋放標記為要重新歸檔的檔案。此指令具有以下格式：

```
research_no_release
```

list_size 指令：調整釋放器候選者清單的大小

您可以使用 `list_size` 指令來指定釋放器候選者的數量。如果您注意到釋放器在釋放必要的檔案數量以達到低標記之前，進行多次檔案系統掃描，則可能需要考慮將該值增大到大於預設值 10,000。包含許多小型檔案的檔案系統可能會發生此情況。您可以從釋放器記錄檔取得有關釋放器作業的資訊。

此指令具有以下格式：

```
list_size = number
```

對於 *number*，請指定 10 到 2,147,483,648 之間的整數。

指定歸檔組中所有檔案的釋放屬性

`archiver.cmd` 檔案中的大部分指令皆會對歸檔造成影響，但歸檔組指定指令則可讓您指定套用至歸檔組中所有檔案的釋放屬性。

[表 4-3](#) 會顯示釋放相關的歸檔組指定值。

表 4-3 釋放用的歸檔組指定值

指令	作用
<code>-release a</code>	指定在製作第一個歸檔備份後，必須釋放歸檔組中的檔案。若要為每個檔案製作多個歸檔備份，請勿使用此選項。在此情況下，系統將分段備份備份 1 以便製作備份 2。
<code>-release d</code>	重設回預設值。
<code>-release n</code>	指定永不釋放歸檔組中的檔案。
<code>-release p</code>	指定在歸檔後必須部分釋放歸檔組中的檔案。

如需有關上述及其他 `archiver.cmd` 指令的更多資訊，請參閱[第 31 頁的「歸檔」](#)。

規劃釋放器作業

為站台確定快取中的檔案特性很有必要。若您只需要分段備份少許 KB，則載入磁帶將十分不符合經濟效益。此時，您可以將系統配置成在快取中保留小型檔案。[程式碼範例 4-5](#) 會顯示 `releaser.cmd` 檔案中用於先釋放最大檔案的指令。

程式碼範例 4-5 先釋放最大檔案的指令

```
weight_size = 1.0
weight_age = 0.0
```

此外，您也可以將最近修改的檔案保留在快取中，因為最近修改的檔案可能很快又會需要修改。此法將可避免檔案分段備份後進行修改時，造成額外的經常性耗用時間。在此情況下，請使用第二組時間權數 (自上次修改之後的時間)。[程式碼範例 4-6](#) 顯示的是 `releaser.cmd` 檔案中用於加權檔案 (從最早修改到最近修改的檔案) 的指令。

程式碼範例 4-6 先釋放最早修改之檔案的指令

```
weight_size = 0.0
weight_age_access = 0.0
weight_age_modify = 1.0
weight_age_residence = 0.0
```

但情況通常不會如此單純。

假設您要先釋放最大的檔案。目前有幾百個大小相同的小檔案，以及幾個大檔案。但釋放器最終仍會釋放所有大檔案。若指定了 `weight_age = 0.0`，釋放器將會依照隨機順序釋放這些小檔案；這是因為這些小檔案的大小一致，所以具有相同的釋放優先權。

在此案例中，您可以設定 `weight_age = 0.01`，以便先釋放大小相同的小型檔案中最舊的一個。

此外，您也可以設定 `weight_size = 1.0` 與 `weight_age = 0.01`。

這些指令將違反最大檔案優先的規則，它會將較小且存取時間較早 (而非較大且存取時間較晚) 的檔案視為較佳候選者。但您可以讓 `weight_age` 小於 `weight_size`，以降低此影響。

例如在之前的設定下，100 分鐘前分段備份的 4 KB 檔案，與剛完成分段備份 8 KB 檔案，都具有相同的釋放優先權。但您若是設定 `weight_age = 0.001`，4 KB 的檔案便須在 1,000 分鐘之前完成分段備份，才能夠和剛完成分段備份的 8 KB 檔案享有相同的優先權。

若要調整優先權權數，可使用 `no_release` 與 `display_all_candidates` 指令手動執行釋放器，以取得依照優先權順序排列的候選者清單。

手動執行釋放器

有時，您可能要手動執行釋放器。因此，您必須知道檔案系統的掛載點以及釋放器所應達到的低標記。

例如，若要在 `/sam1` 檔案系統達到 47% 滿狀態時釋放檔案，請以 `root` 身份登入並鍵入以下指令：

```
# /opt/SUNWsamfs/sbin/sam-releaser /sam1 47 1.0
```

指令行選項會覆寫任何在 `releaser.cmd` 檔案中指定的選項。釋放器執行時，其會將資訊寫入螢幕與釋放器記錄檔 (若已在 `releaser.cmd` 檔案中指定)。如需更多資訊，請參閱「`sam-releaser(1M)` 線上手冊」。

分段備份

分段備份是指將檔案資料從近線或離線儲存裝置複製回線上儲存裝置的程序。

本章將說明 Sun StorageTek SAM 的檔案分段備份功能。本附錄包含以下各節：

- 第 119 頁的「關於 `stager.cmd` 檔案」
- 第 126 頁的「設定預覽請求的優先權」
- 第 129 頁的「計算預覽請求的總優先權」
- 第 129 頁的「設定預覽請求優先機制」

關於 `stager.cmd` 檔案

您可以使用 `stager.cmd` 檔案來指定分段備份程式的運作方式。此檔案的完整路徑名稱是 `/etc/opt/SUNWsamfs/stager.cmd`。分段備份程式的預設運作方式如下：

- 分段備份程式嘗試使用程式庫中的所有磁碟機分段備份檔案。
- 分段備份緩衝區大小由媒體類型決定，分段備份緩衝區未鎖定。
- 未寫入任何記錄檔。
- 最多可同時啟動 1000 個分段備份請求。

`stager.cmd` 檔案可讓您指定覆寫預設運作方式的指令。您可以將分段備份程式配置成立即分段備份檔案、永不分段備份檔案、指定部分分段備份與指定其他分段備份動作。永不分段備份功能用途廣泛，例如從大檔案中隨機存取小記錄的應用程式即可使用此功能；在此功能啟動時，系統將直接從歸檔媒體存取資料，而不先在線上分段備份檔案。

本章節的其餘部分將說明分段備份程式指令。若需關於分段備份程式指令的額外資訊，請參閱「[stager.cmd\(4\) 線上手冊](#)」。

第 125 頁的「[stager.cmd 檔案範例](#)」顯示所有可能的指令均已設定後的已完成 `stager.cmd` 檔案。

備註 – 您若是使用 File System Manager 軟體，即可從 [File System Summary] 或 [File System Details] 頁面控制分段備份。您可以瀏覽檔案系統並查看個別檔案的狀態，使用篩選器檢視特定檔案，以及選取要分段備份的特定檔案。您可以選取分段備份的項目，或由系統自行選擇。

若要設定分段備份程式指令，可使用 `vi(1)` 或其他編輯器編輯 `/etc/opt/SUNWsamfs/stager.cmd` 檔案。接著可使用 `samd(1M)` 指令及其 `config` 選項傳遞檔案變更，然後重新啟動系統：

```
# samd config
```

要取得有關您可以在此檔案中包括的指令之資訊，請參閱以下小節。

- 第 120 頁的「[drives 指令：指定磁碟機數量](#)」
- 第 121 頁的「[設定分段備份緩衝區大小](#)」
- 第 122 頁的「[指定記錄檔](#)」
- 第 125 頁的「[指定分段備份請求數目](#)」

drives 指令：指定磁碟機數量

依據預設，分段備份程式在分段備份檔案時會使用所有可用磁碟機。如果分段備份程式使所有磁碟機都處於忙碌中，則會干擾歸檔器的作業。`drives` 指令可以指定可供分段備份程式使用的磁碟機數目。此指令具有以下格式：

```
drives = library count
```

表 5-1 `drives` 指令的引數

引數	意義
<i>library</i>	程式庫的系列組名稱，如 Sun StorageTek SAM <code>mcf</code> 檔案中所示。
<i>count</i>	表示要使用的最大磁碟機數目。依據預設，此值等於在 <code>mcf</code> 檔案中為此程式庫配置的磁碟機數目。

例如：以下指令行指定只有 dog 系列組程式庫中的一台磁碟機用於分段備份檔案：

```
drives = dog 1
```

如需有關 mcf 檔案的更多資訊，請參閱「mcf(4) 線上手冊」。

您也可以使用 File System Manager 軟體指定此指令。若需更多資訊，請參閱 File System Manager 線上說明。

設定分段備份緩衝區大小

預設會將分段備份的檔案先讀取至記憶體中的緩衝區，然後再從歸檔媒體復原至線上磁碟快取記憶體。您可以使用 `bufsize` 指令指定非預設的緩衝區大小 (選擇使用)，也可以鎖定緩衝區。這些動作可以提高效能，而您也可以嘗試使用不同的 `buffer-size` 值。此指令具有以下格式：

```
bufsize = media buffer-size [lock]
```

表 5-2 `bufsize` 指令的引數

引數	意義
<i>media</i>	指定「mcf(4) 線上手冊」上的清單中列出的歸檔媒體類型。
<i>buffer-size</i>	2 至 1024 的數字。預設值為 4。此值會與該媒體類型的 <code>dev_blksize</code> 值相乘，並使用所得的緩衝區大小。 <code>dev_blksize</code> 值會在 <code>defaults.conf</code> 檔案中指定。 <code>buffer-size</code> 指定的數字愈高，使用的記憶體便愈多。如需有關此檔案的更多資訊，請參閱「 <code>defaults.conf(4)</code> 線上手冊」。
<i>lock</i>	<p><code>lock</code> 引數會指定分段備份歸檔備份時，分段備份程式應使用鎖定的緩衝區。如果指定了 <code>lock</code>，分段備份程式在備份作業過程中會將檔案鎖定在記憶體的分段備份緩衝區內。如此可避免因鎖定及解除鎖定每個 I/O 請求的緩衝區所造成的經常性耗用時間，進而降低系統 CPU 時間。</p> <p><code>lock</code> 引數應該僅在具有大容量記憶體的大系統中指定。記憶體不足會導致出現記憶體不夠用的情況。</p> <p>僅在為要分段備份的檔案啟動了直接 I/O 時，<code>lock</code> 引數才會生效。預設不會指定 <code>lock</code> 參數，且檔案系統會鎖定所有直接 I/O 緩衝區，包括要分段備份的緩衝區。如需有關啓用直接 I/O 的更多資訊，請參閱「<code>setfa(1)</code> 線上手冊」、<code>「sam_setfa(3) 程式庫常式線上手冊」</code> 或 <code>「mount_samfs(1M) 線上手冊」</code> 上的 <code>-O forcedirectio</code> 選項。</p>

您也可以使用 File System Manager 軟體指定此指令。若需更多資訊，請參閱 File System Manager 線上說明。

指定記錄檔

您可以請求 Sun StorageTek SAM 軟體收集檔案分段備份事件資訊，並將其寫入記錄檔中。logfile 指令可以指定分段備份程式可將記錄資訊寫入至其中的記錄檔。此指令具有以下格式：

```
logfile=filename [ event ]
```

關於 *filename*，請指定完整的路徑名稱。

關於 *event*，請指定一或多個分段備份事件。您若是指定了多個事件，請使用空格區隔每個事件。表 5-3 會列出可供指定的事件。

表 5-3 *event* 引數的值

值	動作
all	記錄所有分段備份事件。
start	檔案開始分段備份的時候做記錄。
finish	檔案分段備份結束的時候做記錄。依據預設啟動。
cancel	操作者取消分段備份的時候做記錄。依據預設啟動。
error	記錄分段備份錯誤。依據預設啟動。

指定記錄檔後，分段備份程式會在記錄檔中為每個分段備份過的檔案寫入一或多行資訊。此行所含的資訊包括檔案名稱、分段備份的日期與時間以及磁碟區序號 (VSN) 等。

以下指令行將指定檔案 /var/adm/stage.log：

```
logfile=/var/adm/stage.log
```

程式碼範例 5-1 會顯示分段備份程式記錄檔的範例。

程式碼範例 5-1 分段備份程式記錄檔範例

```
S 2003/12/16 14:06:27 dk disk01 e.76d 2557.1759 1743132 /sam1/testdir0/filebu 1
root other root 0 -
F 2003/12/16 14:06:27 dk disk01 e.76d 2557.1759 1743132 /sam1/testdir0/filebu 1
root other root 0 -
S 2003/12/16 14:06:27 dk disk02 4.a68 1218.1387 519464 /sam1/testdir1/fileaq 1
root other root 0 -
S 2003/12/16 14:06:43 dk disk01 13.ba5 3179.41 750880 /sam1/testdir0/filecl 1
root other root 0 -
F 2003/12/16 14:06:43 dk disk01 13.ba5 3179.41 750880 /sam1/testdir0/filecl 1
root other root 0 -
S 2003/12/16 14:06:59 dk disk01 17.167b 1155.1677 1354160 /sam1/testdir0/filedb
1 root other root 0 -
F 2003/12/16 14:06:59 dk disk01 17.167b 1155.1677 1354160 /sam1/testdir0/filedb
1 root other root 0 -
S 2003/12/16 14:06:59 dk disk02 f.f82 3501.115 1458848 /sam1/testdir1/filecb 1
root other root 0 -
S 2003/12/16 14:07:15 dk disk01 1f.473 1368.1419 636473 /sam1/testdir0/fileed 1
root other root 0 -
S 2003/12/16 14:07:15 dk disk02 16.f15 3362.45 1065457 /sam1/testdir1/filecz 1
root other root 0 -
S 2003/12/16 14:07:31 dk disk01 23.201d 3005.1381 556807 /sam1/testdir0/fileeq
1 root other root 0 -

S 2003/12/16 14:07:47 dk disk01 26.c4d 2831.1113 1428718 /sam1/testdir0/fileez
1 root other root 0 -
S 2003/12/16 14:07:47 dk disk02 1b.835 3736.59 1787855 /sam1/testdir1/filedp 1
root other root 0 -
```

如程式碼範例 5-1 所示，分段備份程式記錄檔含有分成九個欄位的多行資訊。表 5-4 描述分段備份程式記錄檔欄位的內容。

表 5-4 分段備份程式記錄檔欄位

欄位	範例值	內容說明
1	S	分段備份作業 — S 表示啟動，C 表示已取消，E 表示錯誤，F 表示已完成。
2	2003/12/16	以 <i>yyyy/mm/dd</i> 格式表示之分段備份動作的日期。
3	14:06:27	以 <i>hh:mm:ss</i> 格式表示之分段備份動作的時間。
4	dk	歸檔媒體類型。若需有關媒體類型的資訊，請參閱「mcf(4) 線上手冊」。
5	disk01	VSN。
6	e.76d	歸檔檔案在媒體 (tar(1) 檔案) 中的實體開始位置，以及在歸檔檔案中的檔案偏移 (以十六進制格式表示)。
7	2557.1759	Inode 號碼與產生號碼。為表示唯一性，除了 inode 號碼之外，還會使用產生號碼，這是因為 inode 號碼可重複使用。
8	1743132	檔案長度。
9	/sam1/testdir0/filebu	檔案名稱。
10	1	歸檔備份編號。
11	root	檔案的使用者 ID。
12	other	檔案的群組 ID。
13	root	請求者的群組 ID。
14	0	檔案分段備份所在之磁碟機的設備序號。
15	-	A 此欄位中的 v 表示正在為檔案使用資料驗證。

您也可以使用 File System Manager 軟體指定此指令。若需更多資訊，請參閱 File System Manager 線上說明。

指定分段備份請求數目

`maxactive` 指令可讓您指定任何時間均在使用中的分段備份請求數。

此指令具有以下格式：

```
maxactive=number
```

預設 *number* 為 4000。允許的最小數目為 1。最大值為 500,000。

例如，以下指令行指定佇列中同時存在的分段備份請求不能超過 500 個：

```
maxactive=500
```

stager.cmd 檔案範例

程式碼範例 5-2 顯示 `stager.cmd` 檔案範例。

程式碼範例 5-2 `stager.cmd` 檔案範例

```
# This is stager.cmd file /etc/opt/SUNWsamfs/stager.cmd
drives=dog 1
bufsize=od 8 lock
logfile=/var/adm/stage.log
maxactive=500
```

指定歸檔組中所有檔案的分段備份屬性

雖然 `archiver.cmd` 檔案中的大多數指令會影響歸檔，但歸檔組指定指令可讓您指定套用於歸檔組中所有檔案的分段備份屬性。

第 3 章將完整說明歸檔組指定指令及其引數。表 5-5 會顯示可在歸檔組指定指令中使用的分段備份指令。

表 5-5 可在 `archiver.cmd` 檔案中使用的分段備份指令

指令	作用
<code>-stage a</code>	指定應該聯合分段備份歸檔組中的檔案。
<code>-stage d</code>	重設回預設值。
<code>-stage n</code>	指定永不分段備份歸檔組中的檔案。

如需有關上述及其他 `archiver.cmd` 指令的更多資訊，請參閱第 31 頁的「歸檔」。

設定預覽請求的優先權

歸檔器與分段備份程式程序均可請求載入或卸載媒體。如果請求數目超過可用於媒體載入的磁碟機數目，則超過該數目的請求將會傳送至預覽佇列。

預覽佇列中的歸檔與分段備份請求，均為無法立即滿足的請求。依據預設，系統將按先進先出 (FIFO) 順序來滿足預覽請求。

預覽佇列中可以存在的項目數量，決定於 `defaults.conf` 檔案中的 `previews=` 指令。若需有關變更此指令值的資訊，請參閱「`defaults.conf(4)` 線上手冊」。

您可以將不同的優先權指派給預覽請求，亦可在預覽程序檔中輸入指令以覆寫 FIFO 預設值。所輸入的指令將寫入 `/etc/opt/SUNWsamfs/preview.cmd`。

此檔案會根據請求是用於檔案分段備份還是檔案歸檔排定預覽請求。您可增加特定 VSN 的優先權。`preview.cmd` 檔案中的設定亦可根據高標記 (HWM) 或低標記 (LWM) 設定，重新設定所有檔案或特定檔案系統之預覽請求的優先權。

`sam-amld` 常駐程式會在啟動時讀取預覽指令。您必須每行指定一個指令。如果您在 `sam-amld` 常駐程式執行時變更此檔案，您則必須重新啟動 `sam-amld` 常駐程式以使其生效。註解行以井字號 (#) 開頭，延伸至該行的結尾。如需有關此檔案的更多資訊，請參閱「`preview.cmd(4)` 線上手冊」。

以下是可以在 `preview.cmd` 檔案中使用的指令類型：

- 全域指令：適用於所有檔案系統。這些指令必須位於第一個 `fs =` 行的前面。
- 特定於個別檔案系統的指令。這些指令在檔案中必須出現在所有全域指令之後。

檔案系統指令會以 `fs = file-system-name` 開頭。此指令將命名所有後續指令均適用的檔案系統。一個檔案中可能會出現多個檔案指令區段。在下一個 `fs =` 行或檔案結尾之前，檔案系統指令均適用。

備註 – 如有多項指令同時對檔案系統造成影響，則該檔案系統專用的指令將會覆寫全域指令。

全域 VSN 與時間指令

VSN 與時間優先權指令屬於全域指令，因此在 `preview.cmd` 檔案中，會出現在任何檔案系統專用指令之前。

VSN 優先權指令具有以下格式：

```
vsn_priority = value
```

此指令屬於靜態優先權係數，可指定標記為高優先權之 VSN 在總優先權上增加的值。`vsn_priority` 的預設值為 1000.0。當將 VSN 排定為預覽請求時，必須設定其優先權旗標，才可取得此值。請使用 `chmed(1M)` 指令的 `p` 選項來設定優先權旗標 (如 `chmed +p lt.AAA123`)。此旗標對所有針對目前還不是預覽請求之 VSN 而提交的請求均有效。

時間優先權指令具有下列格式：

```
age_priority = factor
```

此指令的整體效果雖屬動態，但卻屬於靜態優先權係數。`age_priority` 係數會與請求成為預覽請求的秒數相乘。所得的結果會與該請求的總優先權相加。等待滿足請求的時間越長，時間係數就越大。設定此係數有助於確保較舊的請求，不會無限制地被具有其他高優先權係數的較新請求取代。

若此係數大於 1.0，便會在計算總優先權時，增加時間係數的優先權。如果此係數小於 1.0，則會減少時間係數的優先權。如果將係數設定為 0.0，則在計算總優先權時會排除時間係數。

未設定優先權旗標的 VSN 之優先權，將隨著它在佇列中的停留時間而增加。其優先權可能會高於以後進入佇列且已設定優先權旗標的 VSN。

全域標記指令或檔案系統專用的標記指令

標記預覽請求指令可用為全域指令或檔案系統專用指令。標記優先權指令可決定預覽請求的標記優先權，如下列方程式所示。

```
lwm_priority +  
lhwm_priority +  
hlwm_priority +  
hwm_priority  
-----  
= water mark priority
```

當標記優先權係數為正數時，計算所得的總優先權結果會增加，致使歸檔請求的優先權高於分段備份請求的優先權。反之，當標記優先權係數為負數時，歸檔請求的總優先權便會降低，致使受理分段備份請求高於歸檔請求。若標記優先權係數為 0.0 (或未指定任何指令)，表示不會執行任何特殊動作。如需更多資訊，請參閱第 130 頁的「範例 1：強制分段備份請求」中的範例。

表 5-6 會顯示四種標記優先權指令及其引數。

表 5-6 標記優先權指令

優先權指令	引數
<code>lwm_priority = value</code>	對於 <i>value</i> ，請指定檔案系統低於 LWM 層級時，歸檔請求之標記優先權係數所應變更的數量。預設值是 0.0。
<code>lhwm_priority = value</code>	對於 <i>value</i> ，請指定檔案系統由低於 LWM 層級而轉為高出該層級，但仍低於 HWM 層級時，歸檔請求之標記優先權係數所應變更的數量。這通常表示檔案系統即將填滿。預設值是 0.0。
<code>hlwm_priority = value</code>	對於 <i>value</i> ，請指定檔案系統由高於 HWM 層級而轉為低於該層級，但仍高於 LWM 層級時，歸檔請求之標記優先權係數所應變更的數量。這通常表示釋放器無法釋放足夠的磁碟空間，以使檔案系統能夠低於 LWM 層級。預設值是 0.0。
<code>hwm_priority = value</code>	對於 <i>value</i> ，請指定檔案系統高於 HWM 層級時，歸檔請求之標記優先權係數所應變更的數量。預設值是 0.0。

這四個標記設定所建立之動態優先權係數包含的百分比，可用於表示檔案系統滿溢程度，以及 HWM 與 LWM 的設定層級。指定給預覽請求的值決定於係數是全域係數、檔案系統特定係數，還是未設定。

檔案系統狀況每有更迭，便會依據適當的標記優先權設定，使用或不使用 `chmed(1M)` 指令的 `p` 選項重新計算與該檔案系統相關之各 VSN 的優先權。

標記優先權只可用於計算歸檔媒體請求。而不用於計算分段備份媒體請求。

程式碼範例 5-3 所顯示的設定可用於啟用釋放器，以釋放足夠的磁碟空間，讓檔案系統得以低於 LWM。

程式碼範例 5-3 降到低於 LWM 的設定

```
lhwm_priority = -200.0
hlwm_priority = 100.0
```

計算預覽請求的總優先權

預覽請求的優先權數值由靜態係數與動態係數共同決定。數字越大，優先權就越高。靜態優先權係數在產生請求時設定。在產生並等待滿足請求後，其結果不會變更總優先權。動態優先權係數在等待滿足請求時可以增加或減少請求的總優先權。

預覽請求的總優先權為所有優先權係數之和。其計算方法如下所示：

```
total priority = vsn_priority + wm_priority + (age_priority *  
time_in_sec_as_preview_request)
```

設定預覽請求優先機制

僅於必要時才變更預設的預覽請求 FIFO 機制，如下列情況所示：

- 確保先處理分段備份請求，然後再處理歸檔請求。
- 確保歸檔請求在檔案系統即將填滿時取得最高優先權。
- 將使用一組指定媒體的請求移至預覽請求清單的頂部。

程式碼範例 5-4 會顯示有提到前述三種情況的 `preview.cmd` 檔案範例。

程式碼範例 5-4 `preview.cmd` 檔案範例

```
# condition 1  
lwm_priority = -200.0  
lhwm_priority = -200.0  
hlwm_priority = -200.0  
# condition 2  
hwm_priority = 500.0  
# condition 3  
age_priority = 1.0
```

對於使用者存取資料非常重要、VSN 磁帶機有限或檔案歸檔做為背景功能執行的環境，您可以使用 `preview.cmd` 檔案來改變儲存設備系統資源為分段備份請求服務的方式。您可以自訂 `preview.cmd` 檔案中的設定，使其支援任何上述的分析藍本及影響已配置的 Sun StorageTek SAM 環境。

由於資料不受本檔案設定的影響，因此在衡量每個預覽請求的優先權時，您可以多方嘗試調整指令設定，以平衡歸檔請求與分段備份請求。

範例 1：強制分段備份請求

下列計算範例將說明如何使用負值的 `wm_priority` 確保分段備份請求的優先權高於歸檔請求。此範例假設：

- 有多個請求在佇列中等待了 100 秒鐘。
- `vsn_priority` 的預設值為 1000。

表 5-7 會顯示請求之總優先權的計算方式。

表 5-7 請求優先權範例

優先	計算
歸檔含優先權的 VSN, LWM :	$1000 + (-200) + (1 \times 100) = 900$
分段備份含優先權的 VSN, LWM :	$1000 + 0 + (1 \times 1100) = 100$
分段備份不含優先權的 VSN, LWM :	$0 + 0 + (1 \times 100) = 100$

範例 2：強制歸檔請求

當在環境中將檔案分段備份回給使用者，以及將新檔案歸檔至媒體兩項作業的重要性平衡為相當時，便應優先考量是否會超出 HWM。在此情況下，若無足夠的檔案符合其歸檔需求，以致無法降低滿檔系統의 百分比，則防止檔案系統滿溢的最佳方法是滿足擱置歸檔請求。

在此情況下，`preview.cmd` 檔案可以非常簡單，如下所示：

```
hwm_priority = 500.0
```

範例 3：依媒體排定請求的優先權

假設您的環境為專案導向，而環境中的某些使用者所處理的檔案群組，是使用特定的 VSN，並與其他使用者相互隔離。在此環境中，某些專案在某些時候可能會具有較高的優先權，因此可能需要從可用系統儲存設備資源取得較高的優先權。您可以使用以下指令來配置 `preview.cmd` 檔案，以讓使用者及其媒體對媒體磁碟機具有適當的優先權：

```
hwm_priority = 5000.0
```

然後，對於優先權使用者群組中的每個 VSN，請輸入以下資訊：

```
# chmed +p lt. VSN
```

此後每一個需要指定 VSN 的請求在預覽佇列中，皆會被置於其他擱置掛載請求之前。

要在以後取消排定使用者媒體的優先權，請為每個 VSN 輸入以下相反的指令：

```
# chmed -p lt. media-type
```

備註 – 如有設定 `chmed(1M)` 指令的 `p` 旗標，則所選 VSN 群組的請求在預覽請求佇列中將具有優先權。

範例 4：複合排定優先權

假設有兩個 Sun StorageTek SAM 檔案系統，它們具有以下請求：

- 所有請求均不應在佇列中等待太久 (`age_priority`)。
- 當其中一個檔案系統低於 LWM 時，分段備份請求即處於優先地位。
- 當其中一個檔案系統超過 LWM，但仍低於 HWM 時，無需對歸檔或分段備份請求排列其優先權。

程式碼範例 5-5 會顯示受影響的指令。

程式碼範例 5-5 指令

```
lwm_priority = -200.0  
lhwm_priority = 0.0  
hlwm_priority = 0.0
```

其他指令不受影響。

如果其中一個檔案系統超出 HWM，歸檔請求即會處於優先地位。

如有兩個檔案系統同時超出 HWM，則防止第二個檔案系統 (如 `samfs2`) 滿溢非常重要。如果 `samfs1` 是使用者工作的檔案系統，而 `samfs2` 是關鍵系統的檔案系統，則更應如此。

程式碼範例 5-6 顯示根據上述清單請求排定請求優先權的 preview.cmd 檔案。

程式碼範例 5-6 顯示複合排定優先權的 preview.cmd 檔案

```
age_priority = 100.0
vsn_priority = 20000.0
lhwm_priority = -200.0
hlwm_priority = -200.0
fs = samfs1
hwm_priority = 1000.0
fs = samfs2
hwm_priority = 5000.0
```

回收

Recycling 是重新恢復歸檔磁碟區空間的程序。回收器使用歸檔器以重新恢復未使用的歸檔備份所佔用的空間。在使用者修改檔案時，與舊版本相關的歸檔備份可從系統中清除。回收器會識別已過期歸檔備份比例最高的磁碟區，並指示將未過期的備份移至不同的磁碟區。如果指定磁碟區僅存在過期備份，則會執行站台定義的動作。例如，磁碟區可以重新製作標籤以立即重新使用，或匯出至遠端儲存裝置，從而保留檔案變更的單獨歷史記錄。使用者不瞭解回收程序，因為它與使用者的資料檔案相關。

本章將說明回收程序與指令。其包含下列主題：

- [第 133 頁的「回收程序簡介」](#)
- [第 135 頁的「回收磁帶櫃」](#)
- [第 143 頁的「回收磁碟歸檔磁碟區」](#)
- [第 145 頁的「回收以實現歸檔備份保留」](#)

回收程序簡介

回收器會將過期的歸檔備份所使用的空間容量，保持為站台指定的參數所定義的最小值。在任何時候，指定歸檔磁碟區的空間由以下部分組成：

- 目前資料，由目前使用中的歸檔影像組成
- 過期資料，由目前已不再使用中的歸檔影像組成
- 可用空間，由目前使用中的歸檔影像或過期歸檔影像未使用的空間組成

磁碟區的容量指磁碟區中資料空間的總容量。例如，10 GB 的磁帶磁碟區中寫入 3 GB 的資料，則其容量是 10 GB，可用空間是 7 GB。

新的或重新製作標籤的歸檔媒體一開始的所有容量均為可用空間。隨著資料歸檔至媒體，可用空間的容量將會減少，而目前資料的容量將會增加。

隨著檔案系統中歸檔檔案的變更或移除，其歸檔影像將會過期並從目前的資料分類移至過期的資料分類。這些影像所使用的實體空間保持不變；僅在檔案系統中不再有檔案指向此空間。僅在空間可回收時，這些影像才能移除，其佔用的空間才能釋放。回收器的目的即是將過期資料所使用的空間轉換成可用空間，而不會遺失任何目前的資料。

回收器和歸檔器會一起運作，詳情如下：

1. 回收器會將磁碟區上存有的所有目前 (有效) 的歸檔影像標示為 `rearchive` 屬性。
2. 如果您要歸檔到可移除的媒體，回收器則會將選定的歸檔磁碟區標示為 `recycle` 屬性。這樣即可防止歸檔器繼續將任何其他歸檔影像寫入磁碟區。
3. 歸檔器會將所有標示的影像移動到另一個磁碟區。此作業就叫作 *rearchiving*。在歸檔器將目前的歸檔影像從舊磁碟區移動到新磁碟區之後，舊磁碟區則只含有可用空間和過期空間。如果您要歸檔到可移除的媒體卡匣，您可以重新製作卡匣標籤及重新使用卡匣。如果您要歸檔到磁碟，回收器則會移除含有過期歸檔影像的檔案。

回收器設計為定期執行。回收器在每次啟動時會執行儘可能多的工作。回收器必須先完成標示重新歸檔的備份，歸檔器才能進行檔案的重新歸檔。

有時設為 `rearchive` 屬性的過期歸檔影像仍會存在於媒體上。在以下情況下可能會發生此情況：

- 歸檔器未在回收器標示過期的歸檔影像之後執行。
- 在移動未過期的歸檔影像時，歸檔器沒有可以使用的媒體。
- 歸檔器還有其他異常現象。

在兩次執行之間，回收器會將狀態資訊保留在程式庫目錄檔與 `inodes` 中。回收程序期間，可以使用 `s1s(1)` 指令及其 `-D` 選項顯示檔案的相關資訊。`s1s(1)` 指令的輸出會顯示檔案是否已排定要進行重新歸檔。

配置回收器之前，請注意下列事項：

- `archiver.cmd` 檔案中的指令可依歸檔集控制回收作業。`recycler.cmd` 檔案中的指令按程式庫控制回收。此外，`recycler.cmd` 檔案控制一般的回收器運作方式。若需有關回收器指令的資訊，請參閱第 135 頁的「使用回收指令」。
- 請勿回收含有可移除的媒體檔案的磁碟區。您是使用 `request(1)` 指令來建立可移除的媒體檔案。具有可移除的媒體檔案的磁碟區將不會耗盡。
- 在 Sun StorageTek SAM 檔案系統上執行維護時，請勿執行回收器。回收器使用 `.inodes` 檔案與 `mcf` 檔案以協助識別目前或過期的檔案，以及與檔案系統相關的裝置。如果 `.inodes` 和 `mcf` 檔案中缺少適當的資訊，則可能導致目前歸檔的資料成為過期資料並被回收。
- 在回收器執行時，所有 Sun StorageTek SAM 檔案系統都必須掛載。如果從線上磁碟回收，包含磁碟區的檔案系統必須已掛載，並且主機系統必須可存取。

注意 – 如果您在具有多部 SAM-QFS 伺服器的環境中使用磁碟歸檔，請務必在配置回收器時格外小心。每部 SAM-QFS 伺服器的 `diskvols.conf` 檔案都必須指向一組唯一的磁碟區資源規格 (磁碟歸檔目標目錄)。如果在不同的 SAM-QFS 伺服器之間共用任何規格，則從其中一部 SAM-QFS 伺服器執行回收器將會銷毀由其他 SAM-QFS 伺服器所管理的磁碟歸檔資料。

回收器依預設未啟用。您必須輸入 `sam-recycler(1M)` 指令才可初始化回收。

回收磁帶櫃

您可以手動或透過 cron(1) 工作輸入 `sam-recycler(1M)` 指令，啓動對磁帶櫃的回收。
表 6-1 顯示多種回收方法。

表 6-1 回收方法和媒體類型

回收方法	媒體和注意事項
使用自動程式庫	可移除的媒體卡匣。 當您使用程式庫進行歸檔時，您是將回收指令輸入到 <code>recycler.cmd</code> 檔案中。
使用歸檔組	可移除的媒體卡匣和磁碟。 當您依歸檔組進行歸檔時，必須將所有回收指令置於 <code>archiver.cmd</code> 檔案中。

如表 6-1 所示，您可以依程式庫或依歸檔集進行回收。如果您要歸檔到磁碟，您就只能使用歸檔組進行回收。

使用回收指令

`recycler.cmd` 檔案接受下列各節中所述的指令：

- 第 135 頁的「指定記錄檔：`logfile` 指令」
- 第 136 頁的「防止回收：`no_recycle` 指令」
- 第 136 頁的「指定回收整個自動程式庫：程式庫指令」
- 第 145 頁的「回收以實現歸檔備份保留」

指定記錄檔：`logfile` 指令

`logfile` 指令用於指定回收器記錄檔。此指令具有以下格式：

```
logfile = filename
```

關於 *filename*，請指定記錄檔的路徑。

以下是 `logfile=` 指令行範例：

```
logfile = /var/adm/recycler.log
```

防止回收：no_recycle 指令

no_recycle 指令用於防止回收磁碟區。要指定 VSN，可使用常規表示式與一個或多個指定的媒體類型。

此指令具有以下格式：

```
no_recycle media-type VSN-regexp [ VSN-regexp ... ]
```

表 6-2 no_recycle 指令的引數

引數	意義
<i>media-type</i>	「mcf(4) 線上手冊」中的媒體類型。
<i>VSN-regexp</i>	一或多個以空格隔開的常規表示式，可用於說明磁碟區。如需相關資訊，請參閱「regexp(5) 線上手冊」或第 59 頁的「使用式樣比對的檔案名稱 search-criterion：-name regexp」。

指定 **media type** 可防止回收特定類型媒體中所儲存的磁碟區。一或多項 *VSN-regexp* 指定可讓您識別要排除在回收作業之外的特定卡匣。

例如，以下指令行排除在回收任何磁帶磁碟區之外，其 VSN 識別標記開頭為 DLT：

```
no_recycle lt DLT.*
```

指定回收整個自動程式庫：程式庫指令

程式庫指令可讓您為指定程式庫相關的 VSN 指定不同的回收參數。

此指令具有以下格式：

```
library parameter [ parameter ... ]
```

對於 *library*，請指定 mcf 檔案之系列集欄位中所指定的程式庫名稱。

對於 *parameter*，請指定表 6-3 中一或多個以空格分隔的 *parameter* 關鍵字。

表 6-3 程式庫指令的 *parameter* 值

參數值	意義
<code>-dataquantity size</code>	回收器在清除磁碟區內有用的資料時，其所能夠排定進行重新歸檔的資料量上限。預設值是 1 GB。
<code>-hwm percent</code>	程式庫高標記。預設值是 95。
<code>-ignore</code>	可防止回收此程式庫中之磁碟區的指令。此指令在測試 <code>recycler.cmd</code> 檔案非常實用。
<code>-mail email-address</code>	回收電子郵件要傳送至的電子郵件地址。依據預設，不會傳送電子郵件。
<code>-mingain value</code>	表示最低 VSN 擷取。預設值是 50。
<code>-vsncount count</code>	回收磁碟區的計數上限。預設值是 1。

例如以下指令行：

```
gr47 -hwm 85 -ignore -mail root -mingain 40
```

該指令行指定了程式庫 `gr47` 的以下內容：

- 在程式庫中的磁碟區耗盡 85% 時，應考慮回收程式庫。
- 最小擷取百分比為 40%。
- 僅回收一個磁碟區。這也是預設設定。
- 回收訊息將透過電子郵件傳送至 `root`。

此外，重新歸檔的資料不可大於 1 GB。此為預設值，因此不會在 `recycler.cmd` 檔案中指定。

配置回收器

初始化回收器時，第 136 頁的「指定回收整個自動程式庫：程式庫指令」中所指定的預設回收器設定便會生效。如需有關釋放器的更多資訊，請參閱「`sam-recycler(1M)` 線上手冊」。

配置磁帶櫃的回收器時，必須依下列順序執行以下各節中所述的作業：

1. 第 138 頁的「建立 `recycler.cmd` 檔案」
2. 第 143 頁的「編輯 `archiver.cmd` 檔案」(此為回收磁帶櫃的選擇性步驟)
3. 第 140 頁的「執行回收器」
4. 第 141 頁的「建立回收器的 `crontab` 檔案」
5. 第 142 頁的「移除 `-recycle_ignore` 和 `ignore` 參數」
6. 第 142 頁的「建立 `recycler.sh` 檔案」

建立 `recycler.cmd` 檔案

若要回收程式庫中卡匣上的歸檔副本，請建立 `recycler.cmd` 檔案。

備註 – 即使要按歸檔組回收，仍然應該在 `recycler.cmd` 檔案中配置每個程式庫。這可確保沒有成為歸檔組的 VSN 在需要時可以回收。

`recycler.cmd` 檔案包含一般回收指令。此檔案還可包含用於 Sun StorageTek SAM 環境中每個程式庫的指令。一般的 `recycler.cmd` 檔案包含以下指令行：

- 一個指定回收器記錄檔的 `logfile=` 指令行。系統會將回收訊息與回收回報寫入此檔案。
- 一個或多個指令行，用於每個包含要回收磁碟區的程式庫。此行必須包含要回收之程式庫的系列組名稱 (在 `mcf` 檔案中)。系列集名稱用於識別回收器的程式庫。若需有關回收指令的資訊，請參閱第 135 頁的「使用回收指令」。

由於您所建立的 `recycler.cmd` 文字行尚未經過測試，因此請使用 `ignore` 關鍵字。您將在此程序的稍後步驟中移除 `ignore` 關鍵字。

▼ 建立 `recycler.cmd` 檔案

1. 成為超級使用者。
2. 使用 `vi(1)` 或其他編輯器來開啟 `/etc/opt/SUNWsamfs/recycler.cmd` 檔案。
3. 新增一或多個本章所述的指令以控制回收器作業。
4. 儲存並關閉檔案。

此外，您也可以使用 File System Manager 軟體建立 `recycler.cmd` 檔案。若需更多資訊，請參閱 File System Manager 線上說明。

`recycler.cmd` 檔案範例

程式碼範例 6-1 顯示 `recycler.cmd` 檔案範例。

程式碼範例 6-1 `recycler.cmd` 檔案範例

```
logfile = /usr/tmp/recycler.log
stk30 -hwm 51 -mingain 60 -ignore -mail root
```

以下各節將說明程式碼範例 6-1 中所指定的參數。

-hwm 51 參數

透過指定高標記，可設定媒體的使用率百分比，低於此百分比時便無法進行回收。此百分比是指程式庫中使用的空間佔其總容量的比率。舉例來說，某程式庫中如有十個 20 GB 的磁帶，而其中三個已經全滿，剩餘七個約使用了 30%，則媒體的使用率百分比將是：

$$((3 * 1.00 + 7 * 0.30) * 20G) / (10 * 20G) * 100\% = 51\%$$

請注意，此計算不區別目前資料與過期資料。僅表示使用的媒體數目。

在此範例中，如果公用程式百分比等於或小於 51%，回收器不會自動選擇回收任何自動程式庫的 VSN。

備註 – 您可以使用下列指令設定回收旗標，以強制回收 VSN：

```
# chmed +c lt.VSN
```

設定 +c 旗標後，歸檔器便不會再將其他歸檔影像寫入該磁碟區中。可透過 samu(1M) 公用程式檢視 +c 旗標。若需更多相關資訊，請參閱「chmed(1M) 線上手冊」和「samu(1M) 線上手冊」。若需關於使用 samu(1M) 公用程式的資訊，請參閱「Sun StorageTek QFS 檔案系統配置與管理指南」。

-mingain 60 參數

最小 VSN 增益百分比用於設定回收卡匣所取得的空間量的下限。例如，某自動程式庫中的卡匣中，95% 為目前資料，5% 為過期資料，則透過回收卡匣所取得的增益將只有 5%。因此不太值得移動其他 95% 的資料以擷取此空間。將最小增益設定為大於或等於 6%，將導致回收器無法自動選取此 VSN。

-ignore 參數

-ignore 參數用於防止回收器回收特定的程式庫。配置回收器時請使用此參數。

-mail 參數

-mail 參數用於指定回收器在對指定的程式庫進行回收時傳送電子郵件。電子郵件訊息具有以下主旨行：

```
Robot robot-name recycle
```

表 6-2 顯示訊息內容範例。

程式碼範例 6-2 回收訊息範例

```
I will recycle VSN vsn.
Cannot find any candidate VSN in this media changer.
Previously selected VSN vsn is not yet finished recycling.
Previously selected VSN vsn is now finished recycling. It will now
be post-recycled.
```

▼ 執行回收器

請遵循下列指示手動執行回收器。

1. 輸入 `sam-recycler(1M)` 指令。
回收器將讀取 `recycler.cmd` 檔案。
2. 針對回收器指示的任何錯誤訊息檢查標準輸出記錄、Sun StorageTek SAM 記錄及 `/var/adm/messages`。
如有錯誤，請更正檔案。

程式碼範例 6-3 顯示回收可移除的媒體卡匣的回收器記錄記錄檔範例。

程式碼範例 6-3 可移除的媒體卡匣的回收器記錄檔範例

```
===== Recycler begins at Wed Dec 12 14:05:21 2001 =====
Initial 2 catalogs:
0 Family: m160 Path: /var/opt/SUNWsamfs/catalog/m160
Vendor: ADIC Product: Scalar 100
SLOT ty capacity space vsn
0 at 25.0G 25.0G CLN005
1 at 48.5G 6.1G 000003
2 at 48.5G 32.1G 000004
3 at 48.5G 35.1G 000005
4 at 48.5G 44.6G 000044
5 at 48.5G 45.1G 000002
6 at 48.5G 45.9G 000033
7 at 48.5G 48.5G 000001
Total Capacity: 364.8G bytes, Total Space Available: 282.3G bytes
```

```

Volume utilization 22%, high 95% VSN_min 50%
Recycling is ignored on this robot.
1 Family: hy                               Path: /var/opt/SUNWsamfs/catalog/historian
Vendor: Sun SAM-FS                         Product: Historian
SLOT          ty      capacity            space vsn
  (no VSNs in this media changer)
Total Capacity: 0      bytes, Total Space Available: 0      bytes
Volume utilization 0%, high 95% VSN_min 50%
Recycling is ignored on this robot.
8 VSNs:

      ---Archives---  -----Percent-----  m160
----Status-----  Count    Bytes    Use  Obsolete  Free  Library:Type:VSN
no-data VSN        0         0         0   87        13   m160:at:000003
no-data VSN        0         0         0   33        67   m160:at:000003
no-data VSN        0         0         0   27        73   m160:at:000005
no-data VSN        0         0         0    8        92   m160:at:000044
no-data VSN        0         0         0    7        93   m160:at:000002
no-data VSN        0         0         0    5        95   m160:at:000033
empty VSN          0         0         0    0        100  m160:at:CLN005
empty VSN          0         0         0    0        100  m160:at:000001
Recycler finished.
===== Recycler ends at Wed Dec 12 14:05:32 2001 =====

```

建立回收器的 crontab 檔案

如果系統按預期執行，則可以為超級使用者建立 crontab 項目，以定期執行回收器。選擇的頻率視站點的條件而定。

如需有關建立 crontab 項目的指示，請參閱「cron(1M) 線上手冊」。

以下會以超級使用者之 crontab 檔案中的項目為範例，以其指定 cron 常駐程式每到奇數小時，即以五分鐘的間隔執行回收器：

```
5 1,3,5,7,9,11,13,15,17,19,21,23 * * * /opt/SUNWsamfs/sbin/sam-recycler
```

▼ 移除 `-recycle_ignore` 和 `ignore` 參數

如果您已使用 `-recycle_ignore` 或 `ignore` 參數在配置期間停用回收，請使用以下程序移除這些參數。除以下程序之外，也可使用 File System Manager 執行此作業。若需更多資訊，請參閱 File System Manager 線上說明。

1. 使用 `vi(1)` 或其他編輯器，從 `archiver.cmd` 檔案中移除 `-recycle_ignore` 參數。
2. 使用 `vi(1)` 或其他編輯器來從 `recycler.cmd` 檔案移除 `ignore` 參數。
現在即開始回收。

建立 `recycler.sh` 檔案

若要回收可移除式媒體卡匣上的歸檔備份，請建立 `recycler.sh` 檔案。如果僅歸檔至磁碟，請不要執行此步驟。

在 VSN 中所有目前影像已重新歸檔至其他 VSN 時，回收器將執行 `recycler.sh` 程序檔。您應決定無需已回收卡匣的站台需求。某些站台選擇為卡匣製作標籤並重新使用卡匣；其他則選擇移除自動程式庫中的卡匣，以稍後用於存取歷史檔案。

回收器會使用以下引數呼叫 `/opt/SUNWsamfs/scripts/recycler.sh` 程序檔：

```
Media type: $1  VSN: $2  Slot: $3  Eq: $4
```

如需程序檔的範例，請參閱「`recycler.sh(1M)` 線上手冊」或 `/opt/SUNWsamfs/examples/recycler.sh`。後者將說明如何重新標記已回收的 VSN，以及如何將郵件傳送給超級使用者。

若需更多相關資訊，請參閱「`recycler(1M)` 線上手冊」和「`recycler.sh(1M)` 線上手冊」。

回收磁碟歸檔磁碟區

配置回收磁碟歸檔磁碟區的回收器時，必須依下列順序執行以下各節中所述的作業：

1. 第 143 頁的「編輯 `archiver.cmd` 檔案」
2. 第 144 頁的「執行回收器」
3. 第 144 頁的「建立回收器的 `crontab` 檔案」
4. 第 144 頁的「移除 `-recycle_ignore` 和 `ignore` 參數」

編輯 `archiver.cmd` 檔案

若要歸檔至磁碟，必須編輯 `archiver.cmd` 檔案，才可進行回收。

若要依程式庫進行回收，可選擇是否要執行此步驟。

若要編輯 `archiver.cmd` 檔案，請遵循第 42 頁的「建立或修改 `archiver.cmd` 檔案及散佈您的變更」中所述的步驟作業。

此外，您也可以使用 File System Manager 編輯 `archiver.cmd` 檔案。若需更多資訊，請參閱 File System Manager 線上說明。

若要啓用依歸檔組進行回收，必須在 `params` 與 `endparams` 指令間增加歸檔組回收指令。表 6-4 會顯示可供使用的歸檔組回收指令。

表 6-4 歸檔組回收指令

指令	功能
<code>-recycle_dataquantity size</code>	限制回收器排程進行重新歸檔的資料量，以清除含有有用資料的磁碟區。依預設，進行磁碟歸檔回收時會忽略此限制。
<code>-recycle_ignore</code>	防止回收歸檔集。
<code>-recycle_mailaddr mail-address</code>	將回收器訊息傳送至指定的電子郵件地址。
<code>-recycle_mingain percent</code>	透過設定磁碟區的 <code>mingain</code> 標記限制歸檔集中磁碟區的回收。 <code>mingain</code> 表示為與磁碟區關聯的過期資料百分比。當磁碟區的過期資料超過 <code>mingain</code> 百分比時，回收器便會開始回收磁碟區。預設值為 50%。
<code>-recycle_minobs percent</code>	透過為回收器的磁碟歸檔磁碟區之重新歸檔程序設定臨界值，限制回收器對磁碟區中 <code>tar</code> 檔案的選取。當磁碟上已歸檔的 <code>tar</code> 檔案中過期檔案百分比達到此臨界值時，回收器便會開始將目前檔案從歸檔移至新的 <code>tar</code> 檔案中。一旦所有目前檔案皆經過移動之後，原始的 <code>tar</code> 檔案便會標記為要從磁碟歸檔中移除的候選者。預設值為 50%。
<code>-rearch_stage_copy copy-number</code>	允許分段備份，以從所選的 (速度較快的) 備份執行重新歸檔。

如需有關歸檔器指令的更多資訊，請參閱第 3 章或「`archiver.cmd(4)` 線上手冊」。

▼ 執行回收器

請遵循下列指示手動執行回收器。

1. 輸入 `sam-recycler(1M)` 指令。
回收器將讀取 `recycler.cmd` 檔案。
2. 針對回收器指示的任何錯誤訊息檢查標準輸出記錄、**Sun StorageTek SAM** 記錄及 `/var/adm/messages`。
如有錯誤，請更正檔案。

程式碼範例 6-4 顯示的是用於回收磁碟歸檔檔案的回收器記錄檔範例。

程式碼範例 6-4 磁碟歸檔檔案的回收器記錄檔範例

```
---Archives---  -----Percent-----  
----Status-----      Count   Bytes   Use Obsolete Free   Library:Type:VSN  
new candidate          0         0     0    41     59 <none>:dk:disk01  
677 files recycled from VSN disk01 (mars:/sam4/copy1)  
0 directories recycled from VSN disk01 (mars:/sam4/copy1)
```

建立回收器的 crontab 檔案

如果系統按預期執行，則可以為超級使用者建立 `crontab` 項目，以定期執行回收器。頻率視站點的條件而定。

如需有關建立 `crontab` 項目的指示，請參閱「`cron(1M)` 線上手冊」。

以下會以超級使用者之 `crontab` 檔案中的項目為範例，以其指定 `cron` 常駐程式每到奇數小時，即以五分鐘的間隔執行回收器：

```
5 1,3,5,7,9,11,13,15,17,19,21,23 * * * /opt/SUNWsamfs/sbin/sam-recycler
```

▼ 移除 `-recycle_ignore` 和 `ignore` 參數

如果您已使用 `-recycle_ignore` 或 `ignore` 參數在配置期間停用回收，請使用以下程序移除這些參數。除以下程序之外，也可使用 **File System Manager** 執行此作業。若需更多資訊，請參閱 **File System Manager** 線上說明。

1. 使用 `vi(1)` 或其他編輯器，從 `archiver.cmd` 檔案中移除 `-recycle_ignore` 參數。
2. 使用 `vi(1)` 或其他編輯器來從 `recycler.cmd` 檔案移除 `ignore` 參數。
現在即開始回收。

回收以實現歸檔備份保留

做為一般回收程序的替代方法，可以使用 `sam-nrecycler(1M)` 工具結合 `File System Manager` 的備份與回復點功能進行作業。此工具會移除過期的歸檔備份並釋放歸檔磁碟區，以協助使用 `Sun StorageTek SAM` 傾印檔案以保留歸檔的功能。若要使用此功能，您必須使用此回收器取代現有的 `sam-recycler` 指令。

`sam-nrecycler(1M)` 工具會掃描檔案系統中介資料和 `Sun StorageTek SAM` 傾印檔案，以確定包含歸檔影像的可移除式媒體和磁碟歸檔磁碟區。在非尖峰時段透過 `crontab(1)` 檔案，或在任何時候使用 `sam-nrecycler` 指令可呼叫此工具。`nrecycler` 會透過掃描所有檔案系統的 `.inodes` 檔案和指定的 `Sun StorageTek SAM` 傾印檔案，識別可移除式媒體磁碟區上或磁碟歸檔 `tar` 檔案中所存在的全部歸檔影像。透過掃描檔案系統和 `Sun StorageTek SAM` 傾印檔案，`nrecycler` 可確定是否存在不包含任何歸檔影像的磁碟區，並收回這些磁碟區的空間。如果可移除式媒體磁碟區不包含任何歸檔影像，則可安全地重新標記卡匣。如果磁碟歸檔 `tar` 檔案不包含任何歸檔影像，則可安全地從磁碟歸檔目錄移除 `tar` 檔案。

必須使用 `/etc/opt/SUNWsamfs/nrecycler.cmd` 檔案提供 `sam-nrecycler(1M)` 的指令。您必須指定包含要搜尋的 `Sun StorageTek SAM` 傾印檔案之目錄的路徑。如果指令檔中未指定目錄，則不會進行回收。目錄清單必須完整且包含所有 `Sun StorageTek SAM` 傾印檔案。

您也可以將 `nrecycler.cmd` 檔案中納入 `logfile=` 指令行，以指定 `nrecycler` 記錄檔。系統會將回收訊息與回收回報寫入此檔案。

當 `sam-nrecycler(1M)` 偵測到可移除式媒體磁碟區僅包含可用空間或過期空間，並且可安全重新標記時，會呼叫 `sam-nrecycler.sh` 程序檔。此程序檔可使用原始 `VSN` 或新 `VSN` 重新標記卡匣。然後，程序檔將會從程式庫中匯出卡匣，或執行其他使用者定義的動作。

當 `sam-nrecycler` 偵測到磁碟歸檔磁碟區僅包含可用空間或過期空間時，將會取消連結未使用的磁碟歸檔 `tar` 檔案。

如需有關 `sam-nrecycler(1M)` 的更多資訊，請參閱「`sam-nrecycler(1M)` 線上手冊」。

第7章

使用 Sun SAM-Remote 軟體

Sun SAM-Remote 用戶端與 Sun SAM-Remote 伺服器形成了用戶端/伺服器實作，允許在 Sun StorageTek SAM 主機系統之間共用程式庫及其他可移除式媒體裝置。Sun SAM-Remote 可讓您配置歸檔與分段備份中央程式庫或磁光碟庫中檔案的多個儲存用戶端。例如：如果您的主機系統所在網路跨越廣闊的地理區域，您可以將在某個城市建立的檔案歸檔至遠在異地的程式庫中的卡匣。

本章包括以下各節：

- [第 147 頁的「Sun SAM-Remote 軟體簡介」](#)
- [第 152 頁的「配置 Sun SAM-Remote 軟體」](#)
- [第 164 頁的「使用 Sun SAM-Remote 軟體進行回收」](#)

Sun SAM-Remote 軟體簡介

本簡介中涵蓋下列主題：

- [第 148 頁的「功能」](#)
- [第 149 頁的「需求」](#)
- [第 149 頁的「限制」](#)
- [第 150 頁的「用戶端與伺服器的互動」](#)

功能

圖 7-1 顯示了配置有兩台 Sun SAM-Remote 主機系統伺服器的環境。每部伺服器均具有兩個用戶端。

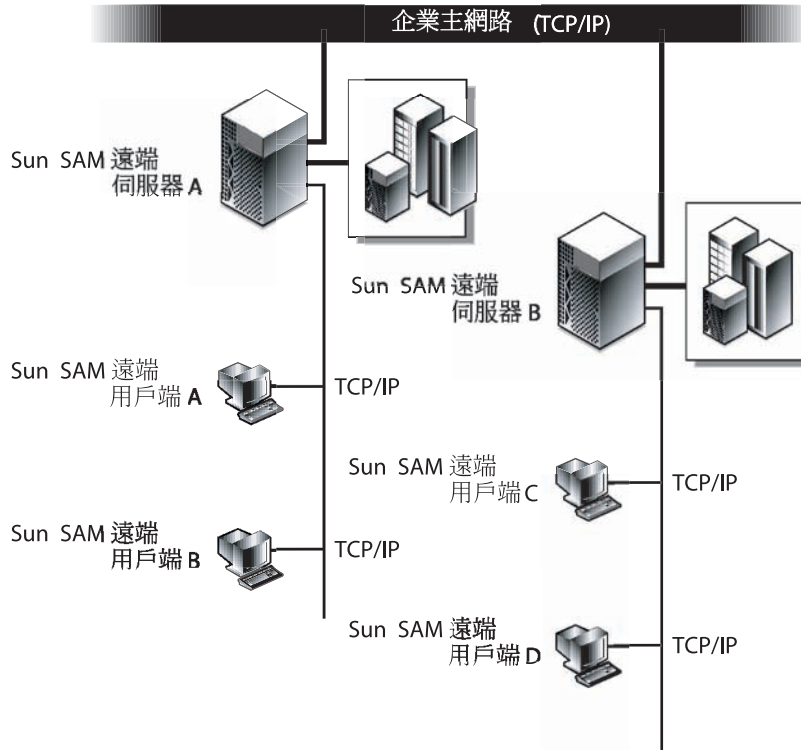


圖 7-1 Sun SAM-Remote 伺服器與用戶端

Sun SAM-Remote 軟體具有下列優點：

- 可讓您在兩個或多個 Sun SAM-Remote 用戶端之間配置遠端共用昂貴的可移除式媒體資源 (如程式庫)。
- 可讓用戶端將資料移至伺服器。
- 啟用多重 Sun StorageTek SAM 伺服器使其成為彼此的主機。在 Sun SAM-Remote 環境中，伺服器是指在 mcf 檔案中設備類型配置為 ss 的主機系統。

您可以配置 Sun SAM-Remote 伺服器與用戶端，以在兩個或多個 Sun Solaris 主機系統之間提供多個歸檔備份。例如，您可將執行 Sun StorageTek SAM 軟體的兩個 Solaris 系統配置為彼此的 Sun SAM-Remote 伺服器與 Sun SAM-Remote 用戶端。此配置的優點之一，就是可以為在其他伺服器上具有資料額外歸檔備份的每台伺服器建立本機備份。您可以使用標準 NFS 在伺服器之間共用檔案系統。在對本機程式庫的存取中斷時，Sun SAM-Remote 軟體會自動從歸檔備份擷取檔案資料。因此，即使主要儲存程式庫無法使用，伺服器使用者仍可繼續存取其資料。

需求

嘗試配置 Sun SAM-Remote 環境之前，請確定您的環境包含下列軟體與硬體：

- 具有已授權、已安裝且可作業的 Sun StorageTek SAM 4U0 或較新的儲存與歸檔管理套裝軟體的 SPARC® 或 x64 系統。
- 具有相同的 Sun StorageTek SAM 軟體版本等級和相同已安裝之修補程式集的主機系統。如需升級某些主機系統，請參閱「Sun StorageTek Storage Archive Manager 安裝與升級指南」。
- 一個或多個做為 Sun SAM-Remote 伺服器的主機系統，並且在其之上至少安裝有一個 SAM-QFS 檔案系統。
- 網路連線在用戶端和安裝 Sun StorageTek SAM 軟體的伺服器間執行 TCP/IP 連線。

限制

Storage Archive Manager 會以相同方式處理遠端程式庫與本機程式庫中的卡匣。但是，下列資訊指明了 Sun SAM-Remote 軟體的限制：

- 您可以使用 Sun SAM-Remote 回收媒體，但只能在徹底測試您的環境後，才嘗試執行此操作。如需更多資訊，請參閱第 164 頁的「[使用 Sun SAM-Remote 軟體進行回收](#)」。
- Sun SAM-Remote 用戶端上只有一個常駐程式可以與 Sun SAM-Remote 伺服器通訊。
- 因此，Sun StorageTek SAM 軟體和 SAM-Remote 無法在共用的 Sun StorageTek QFS 檔案系統中的 Sun StorageTek QFS 用戶端上運作。如果 Sun StorageTek SAM 軟體和 SAM-Remote 執行所在的伺服器既是某些 Sun StorageTek QFS 檔案系統之中介資料伺服器，又兼做其他 Sun StorageTek QFS 檔案系統之用戶端時，則它們僅會在以該伺服器做為中介資料伺服器的檔案系統上運作。

用戶端與伺服器的互動

Sun SAM-Remote 用戶端使用 TCP/IP 連線與 Sun SAM-Remote 伺服器互動。Sun SAM-Remote 用戶端之間的網路可以是 Solaris 作業系統所支援的任何網路類型，如乙太網路、快速乙太網路或光纖通道。

圖 7-2 顯示了 Sun SAM-Remote 用戶端與 Sun SAM-Remote 伺服器之間的互動。

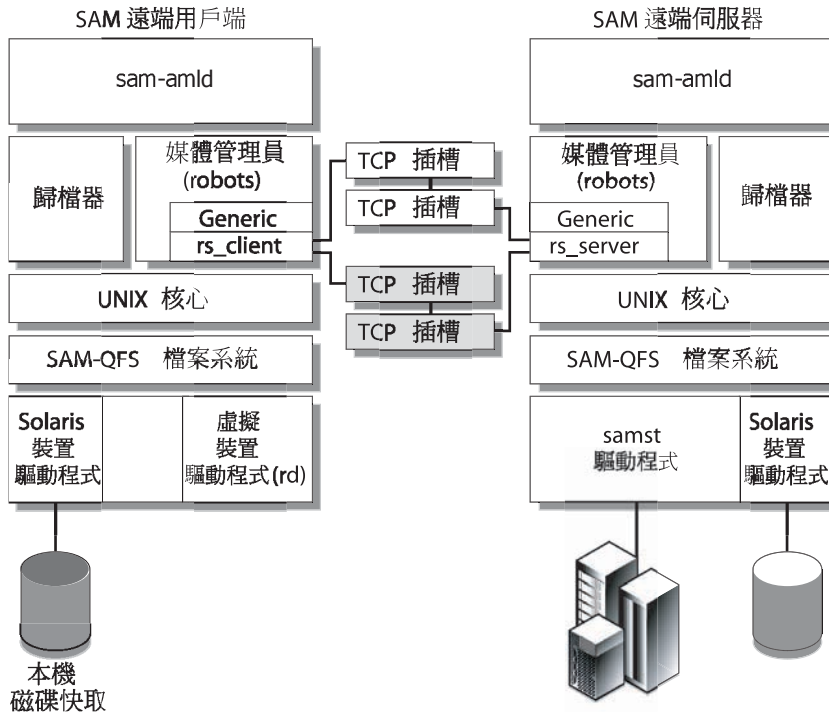


圖 7-2 Sun SAM-Remote 伺服器與用戶端之間的互動

Sun SAM-Remote 伺服器簡介

Sun SAM-Remote 伺服器組成包括全容量 Sun StorageTek SAM 儲存管理主機，以及定義用戶端間共用的程式庫之 Sun SAM-Remote 伺服器常駐程式。Sun SAM-Remote 伺服器上須至少安裝一個 SAM-QFS 檔案系統。

在伺服器系統的 `/etc/opt/SUNwsamfs/mcf` 檔案中新增一行，並將設備類型設定為 `ss`，即可將主機系統定義為 Sun SAM-Remote 伺服器。您必須為每台伺服器提供一個唯一的系列組名稱。您最多可以為每個常駐程式配置十個用戶端。若要配置十個以上的用戶端，請在 `mcf` 檔案中針對所要配置的十個用戶端，各增加一項額外的遠端伺服器項目。如需有關伺服器常駐程式的更多資訊，請參閱「`sam-remote(7)` 線上手冊」。

Sun SAM-Remote 用戶端簡介

Sun SAM-Remote 用戶端為 Sun StorageTek SAM 主機系統，其建立包含虛擬裝置數目的 Sun SAM-Remote 用戶端常駐程式。

在用戶端系統的 `/etc/opt/SUNWsamfs/mcf` 檔案中新增一行，並將設備類型設定為 `sc`，即可將主機系統定義為 Sun SAM-Remote 用戶端。如需有關用戶端常駐程式的更多資訊，請參閱「[sam-remote\(7\) 線上手冊](#)」。

虛擬裝置可定義 Sun SAM-Remote 伺服器上實際可移除的媒體裝置的網路連線。虛擬裝置的設備類型為 `rd`，即 *remote device* (遠端裝置) 的縮寫。您可以在 Sun SAM-Remote 用戶端的 `/etc/opt/SUNWsamfs/mcf` 檔案中定義虛擬裝置。Sun SAM-Remote 常駐程式與虛擬裝置均與一台特定伺服器相關聯。

Sun SAM-Remote 常駐程式支援每個用戶端的虛擬裝置數目並無限制。用戶端使用的實際虛擬裝置數目是可以配置的。在確定應該為每個用戶端配置多少個虛擬裝置時，請將其配置為用戶端與伺服器之間的同步資料傳輸數。因為定義的虛擬裝置越多，網路流量總負載可能會越大。作為系統管理員，您可以自行決定系統所需的實際虛擬裝置數目。

Sun SAM-Remote 伺服器與 Sun SAM-Remote 用戶端之間的互動

Sun SAM-Remote 伺服器常駐程式，`sam-serverd` 偵聽連接埠 1000 上的用戶端。您可以在配有 `rmtsam` 服務名稱的 Sun Solaris `/etc/services` 目錄上配置不同的連接埠。在 Sun SAM-Remote 用戶端連線到 Sun SAM-Remote 伺服器時，`sam-serverd` 常駐程式會在另一個連接埠上建立連線，並使用所定義的連接埠將此連接埠號碼傳送到該用戶端。通訊端大小也會傳送到該用戶端。通訊端大小可加以配置，這在[第 152 頁的「配置 Sun SAM-Remote 軟體」](#)中有詳細的說明。

程式庫目錄

Sun SAM-Remote 程式庫目錄是 Sun SAM-Remote 伺服器上的目錄子集。用戶端目錄檔會即時更新。只有 Sun SAM-Remote 伺服器才能控制分配給 Sun SAM-Remote 用戶端目錄檔的插槽。

初始化之後，系統會根據 Sun SAM-Remote 伺服器目錄檔中的資訊建立用戶端目錄檔，並將其傳送到 Sun SAM-Remote 用戶端。在主機與用戶端之間建立連線後，會將用戶端可以使用的媒體標記為可用。如果用戶端與伺服器之間的連線中斷，則會將用戶端上的媒體標記為無法使用。您可以透過 `samu(1M) v` 顯示來檢視媒體可用性。用戶端上的 `samu(1M) v` 顯示中出現的資訊，是伺服器上 `v` 顯示中出現的資訊子集。因此，您應透過 Sun SAM-Remote 伺服器上的 `samu(1M) v` 畫面存取媒體目錄檔。如需有關 Sun SAM-Remote 伺服器用戶端檔案的更多資訊，請參閱[第 152 頁的「配置 Sun SAM-Remote 軟體」](#)。若需關於使用 `samu(1M)` 公用程式的資訊，請參閱「[Sun StorageTek QFS 檔案系統配置與管理指南](#)」。

如有必要，系統將在主機之間傳送對目錄檔所作的變更。任何伺服器目錄檔中涉及用戶端相關媒體類型的變更，均會傳送到用戶端，並在隨後更新用戶端目錄檔。

歸檔

Sun SAM-Remote 歸檔程序與 Sun StorageTek SAM 的歸檔程序相同。Sun SAM-Remote 用戶端會使掛載請求新增到伺服器的掛載請求表。然後，用戶端會等待伺服器傳回指明媒體是否已掛載的回應訊息。只要媒體可以使用，便會開始歸檔程序。

配置 Sun SAM-Remote 軟體

本節說明如何執行 Sun SAM-Remote 伺服器與用戶端軟體的初始配置。其包含下列章節：

- 第 152 頁的「配置範例」
- 第 153 頁的「配置軟體」

配置範例

圖 7-3 說明的是本教程中所使用的配置範例。本章中的範例將說明如何配置稱為 chicago 的 Sun SAM-Remote 伺服器。

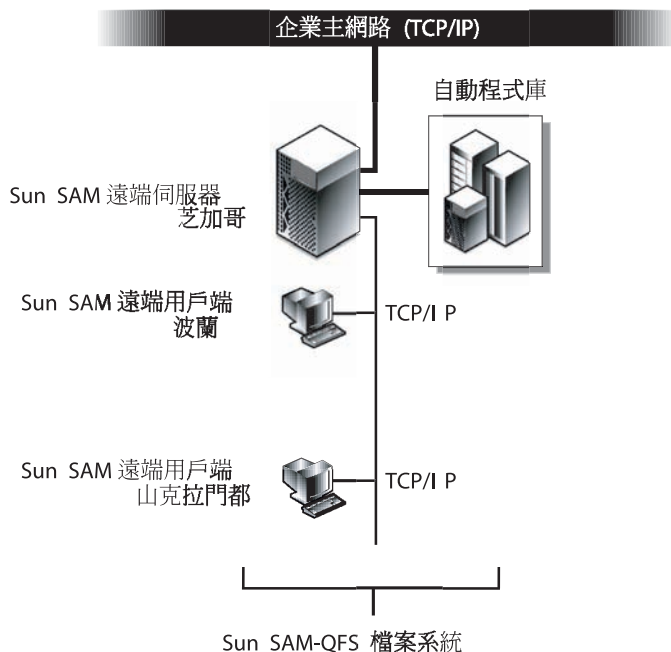


圖 7-3 Sun SAM-Remote 配置範例

在 portland 和 sacramento 上的 Sun StorageTek SAM 檔案系統使用 chicago 做為 Sun SAM-Remote 伺服器。

配置軟體

下列程序將說明如何在 Sun SAM-Remote 伺服器，以及一個或多個 Sun SAM-Remote 用戶端上配置 Sun SAM-Remote 軟體。這些程序必須以下列各節所述之順序執行：

1. [第 153 頁的「登入潛在的伺服器與用戶端主機」](#)
2. [第 153 頁的「確認用戶端與伺服器配置」](#)
3. [第 155 頁的「編輯 mcf 檔案」](#)
4. [第 157 頁的「定義 Sun SAM-Remote 用戶端」](#)
5. [第 157 頁的「在伺服器的 mcf 檔案中定義 Sun SAM-Remote 伺服器」](#)
6. [第 158 頁的「建立 Sun SAM-Remote 伺服器配置檔案」](#)
7. [第 161 頁的「啓用歸檔」](#)

▼ 登入潛在的伺服器與用戶端主機

針對可能使用的伺服器與所有可能使用的用戶端主機執行下列步驟。

1. 以超級使用者的身份登入 **Sun SAM-Remote** 伺服器。

您必須對要安裝 Sun SAM-Remote 軟體的伺服器系統具有超級使用者權限。

2. 以超級使用者的身份登入 **Sun SAM-Remote** 用戶端。

您必須對要安裝 Sun SAM-Remote 軟體的用戶端系統具有超級使用者權限。

▼ 確認用戶端與伺服器配置

為確保您安裝了所需要的軟體等級，請在要配置為 Sun SAM-Remote 環境一部分的各個系統上執行下列步驟。

1. 輸入 `pkginfo(1M)` 指令及其 `-l` 選項，並檢查輸出。

程式碼範例 7-1 會顯示 pkginfo(1M) 指令的輸出範例。

程式碼範例 7-1 使用 pkginfo(1)

```
chicago# pkginfo -l SUNWsamfs
  PKGINST:  SUNWsamfs
    NAME:  Sun SAM-FS and Sun SAM-QFS software Solaris 2.8
  CATEGORY:  system
    ARCH:  sparc
  VERSION:  4.0.5,REV=5.8.2003.01.12
    VENDOR:  Sun Microsystems, Inc.
    PSTAMP:  boomerang-20020712183351
  INSTDATE:  Jan 20 2003 07:30
    HOTLINE:  Please contact your local service provider
    STATUS:  completely installed

  FILES:    489 installed pathnames
           12 shared pathnames
            1 linked files
           51 directories
          179 executables
          35813 blocks used (approx)

chicago#
```

您必須在要配置為 Sun SAM-Remote 環境之一部分的所有用戶端與伺服器主機上，均安裝相同層級的 Sun StorageTek SAM 發行版本和修訂版本。

在程式碼範例 7-1 所示的輸出範例中，會顯示伺服器 (chicago) 所執行的軟體版本為 4U0.5，因此該伺服器所在環境中的所有系統亦必須執行 4U0.5。

2. 輸入 showrev(1M) 指令及其 -p 選項，並檢查輸出。

程式碼範例 7-2 會顯示 showrev(1M) 指令的輸出範例。

程式碼範例 7-2 使用 showrev(1M)

```
chicago# showrev -p | grep SUNWsamfs
Patch: 113546-07 Obsoletes: Requires: Incompatibles: Packages:
SUNWsamfs
chicago#
```

您必須在要配置為 Sun SAM-Remote 環境之一部分的所有用戶端與伺服器主機上，均安裝相同的修補程式集。

在程式碼範例 7-2 所示的輸出範例中，會顯示伺服器執行的修補程式為 113546-07，因此該伺服器所在環境中的所有系統亦必須執行修補程式 113546-07。

如有部分系統是執行舊版的軟體或修補程式，則必須將所有系統升級至最新的軟體層級。使用[程式碼範例 7-1](#) 做為範例，如果您正在任何系統上執行早於版本 4U0.5 的 Sun StorageTek SAM 版本，您至少必須升級至 4U0.5。

若需有關執行軟體升級的資訊，請參閱「Sun StorageTek Storage Archive Manager 安裝與升級指南」。

▼ 編輯 mcf 檔案

1. 從 Sun SAM-Remote 伺服器停止 Sun StorageTek SAM 功能。

- a. 輸入 `samcmd(1M)` 指令及其 `idle eq` 選項，以閒置受 Sun StorageTek SAM 軟體控制的每個可移除式媒體磁碟機。

```
# samcmd idle eq
```

對於 `eq`，請依據 mcf 檔案所定義，指定所要定址之可移除式媒體磁碟機的設備序號。

如需有關 `samcmd(1M)` 指令的更多資訊，請參閱「`samcmd(1M)` 線上手冊」。

或者，您也可以使用 `samu(1M)` 運算子公用程式來閒置磁碟機。若需關於使用 `samu(1M)` 公用程式的資訊，請參閱「Sun StorageTek QFS 檔案系統配置與管理指南」。

備註 – 在您執行 `samd stop` 指令之前，應該先閒置 Sun StorageTek SAM 環境中的磁碟機。這將允許歸檔器、分段備份程式及其他程序完成目前作業，同時也允許系統卸載卡匣，並將卡匣插入其儲存插槽。

- b. 輸入 `samd(1M)` 指令及其 `stop` 選項，以停止 `sam-initd` 常駐程式及其子程序。

```
# samd stop
```

`samd(1M)` 指令安裝在 `/opt/SUNWsamfs/sbin` 中。

2. 在每一個用戶端上使用 `vi(1)` 或其他編輯器編輯現有的 Sun StorageTek SAM `/etc/opt/SUNWsamfs/mcf` 檔案。

此步驟的目標是要將主機定義為 Sun SAM-Remote 用戶端。[程式碼範例 7-3](#) 表示用戶端 `portland` 上已編輯的 mcf 檔案。mcf 檔案將定義檔案系統，並顯示 Sun SAM-Remote 用戶端 `portland` 正被定義為 Sun SAM-Remote 伺服器 `chicago`。

在本章的範例中，您必須為用戶端 `sacramento` 完成相同的配置程序。對於此系統，編輯 `mcf` 檔案並從 `portland` 的 `mcf` 檔案複製行的最後一組至 `sacramento` 的 `mcf` 檔案。這些行可將 `chicago` 主機定義為 `Sun SAM-Remote` 用戶端。

程式碼範例 7-3 portland 上的 mcf 檔案

```
# mcf file on portland
#
# Sun StorageTek QFS file system
#
# Equipment          Eq   Eq   Family      Dev  Additional
# Identifier          Ord  Ty   Set         St   Parameters
# =====          ===  ==  =====    ==  =====
samfs1                1   ms   samfs1      on
/dev/dsk/c1t1d0s0    10  md   samfs1      on  /dev/rdisk/c1t1d0s0
/dev/dsk/c1t2d0s0    12  md   samfs1      on  /dev/rdisk/c1t2d0s0
#
# Define Sun SAM-Remote Client portland to Sun SAM-Remote server chicago
#
/etc/opt/SUNWsamfs/rmt200 200  sc  chicagoss  on  /var/opt/SUNWsamfs/catalog/tcat
/dev/samrd/rd0          201  rd  chicagoss  on
/dev/samrd/rd1          202  rd  chicagoss  on
```

用戶端上的 `mcf` 項目組成包括 `Sun SAM-Remote` 用戶端的一行項目，以及要配置的每個裝置之虛擬裝置項目。這些項目會採用「`mcf(4)` 線上手冊」中所定義的語法。

第一組項目定義 `Sun StorageTek QFS` 檔案系統。

第二組項目將 `Sun SAM-Remote` 用戶端 (`portland`) 定義為 `Sun SAM-Remote` 伺服器 (`chicago`)。第一行定義 `Sun SAM-Remote` 伺服器本身，如下所示。

- 設備識別碼；由用戶端配置檔案的路徑名稱組成，該名稱稍後將於第 157 頁的「[定義 Sun SAM-Remote 用戶端](#)」中建立。
- 設備序號；由 1 至 65535 之間之唯一號碼所構成。
- 設備類型；由兩個字母的快捷字元所構成，可用於識別 `Sun SAM-Remote` 用戶端。
- 系列組；由此特定伺服器所使用之常駐程式的系列組名稱組成。`Sun SAM-Remote` 伺服器的每個用戶端可以具有一個伺服器常駐程式。
- 裝置狀態；可為開啓或關閉。
- 其他參數；在此是指由目錄檔的路徑組成的可選擇欄位。

此 `mcf` 檔案的最後兩個項目會定義 `Sun SAM-Remote` 虛擬裝置，如 `rd` 設備類型所指示。虛擬裝置可定義 `Sun SAM-Remote` 伺服器上實際裝置的網路連線。虛擬裝置項目會在系統重新開機時建立。您可以定義虛擬裝置的無限制數目。

▼ 定義 Sun SAM-Remote 用戶端

Sun SAM-Remote 用戶端之配置檔案中所含的單行項目，是由 Sun SAM-Remote 伺服器名稱所組成。如步驟 2 中的第 155 頁的「編輯 mcf 檔案」所示，此用戶端配置檔案的完整路徑名稱是在用戶端的 mcf 檔案中指定的。

1. 在每一個用戶端上使用 vi(1) 或其他編輯器，開啟名為 **Sun SAM-Remote 用戶端配置檔案** 的檔案。

例如：

```
portland# vi /etc/opt/SUNWsamfs/rmt200
```

2. 編輯該檔案並只包含 **Sun SAM-Remote 伺服器名稱**。

程式碼範例 7-4 顯示於 portland 編輯後的用戶端配置檔案。其指向稱為 chicago 的 Sun SAM-Remote 伺服器。

程式碼範例 7-4 用戶端配置檔案

```
portland# cat /etc/opt/SUNWsamfs/rmt200
chicago
```

▼ 在伺服器的 mcf 檔案中定義 Sun SAM-Remote 伺服器

備註 – 您必須至少有一個於 Sun SAM-Remote 伺服器的 mcf 檔案中已配置之 Sun StorageTek SAM 檔案系統。

- 在 **Sun SAM-Remote 伺服器** 上使用 vi(1) 或其他編輯器，編輯現有的 **Sun StorageTek SAM /etc/opt/SUNWsamfs/mcf 檔案**。

假設您編輯伺服器 chicago 上的 mcf 檔案。所產生的檔案會定義 Sun StorageTek QFS 檔案系統，並將 chicago 定義為 Sun SAM-Remote 伺服器。

程式碼範例 7-5 顯示在 chicago 上的 mcf 檔案。

程式碼範例 7-5 chicago 上的 mcf 檔案

```
# mcf file on Sun SAM-Remote server chicago:
# Eq Identifier Eq Ord  Eq Typ Fam Set Dev St  Addl Params
#
samfs1          1    ms    samfs1 on
/dev/dsk/c2t6d0s0 11  md    samfs1 on /dev/rdisk/c2t6d0s0
/dev/dsk/c2t6d0s1 12  md    samfs1 on /dev/rdisk/c2t6d0s1
#
# define a tape library that client portland can use:
/dev/samst/c0t3u0 100  rb    rb100  on /var/opt/SUNWsamfs/catalog/rb100.cat
/dev/rmt/0cbn    101  tp    rb100  on
/dev/rmt/1cbn    102  tp    rb100  on
# Define Sun SAM-Remote server chicago
#
/etc/opt/SUNWsamfs/rmt200 50  ss      chicagoss on
```

項目語法在第 155 頁的「編輯 mcf 檔案」中有所說明。

▼ 建立 Sun SAM-Remote 伺服器配置檔案

Sun SAM-Remote 伺服器配置檔案可定義磁碟緩衝區特性，以及用於每個用戶端的媒體。您可以為每個伺服器常駐程式配置十個用戶端。若要支援更多的用戶端，必須依照之前在第 155 頁的「編輯 mcf 檔案」(步驟 2) 及第 157 頁的「定義 Sun SAM-Remote 用戶端」中所述，配置另一個 Sun SAM-Remote 伺服器常駐程式。

程式碼範例 7-6 顯示的是位於 Sun SAM-Remote 伺服器 chicago 上的伺服器配置範例檔案 `/etc/opt/SUNWsamfs/rmt200`。此檔案將定義用戶端 portland 與 sacramento。

1. 在伺服器上，使用 `vi(1)` 或其他編輯器開啟稱為 **Sun SAM-Remote 伺服器配置檔案** 的檔案。
2. 編寫伺服器配置檔案。

如程式碼範例 7-6 所示，伺服器配置檔案是由每個用戶端的多行項目所組成。井字號 (#) 表示註解行。註解行右側的內容均可忽略。

程式碼範例 7-6 伺服器配置檔案 rmt200

```
#
# Sun SAM-Remote server config file /etc/opt/SUNWsamfs/rmt200
#
portland
  media
    100 at (000031|000032|000034|000035|000037|000038)
  endmedia
#

sacramento
  media
    100 at (000131|000132|000134|000135|000137|000138)
  endmedia
```

程式碼範例 7-7 顯示的是 Sun SAM-Remote 伺服器配置檔案的格式。

程式碼範例 7-7 伺服器配置檔案格式

```
client-name
  [ parameter1 ]
media
    eq media-type regex
    [ eq media-type regex ]
    [ . . . ]
endmedia
```

a. 編寫 *client-name* 欄位。

client-name 將定義此呼叫 Sun SAM-Remote 常駐程式所適用的每個用戶端網路名稱。 *client-name* 的第一個字元必須是該行的第一個字元。 *client-name* 可以指定為網路名稱、IP 位址或完全合格的網域名稱。

自 *client-name* 之後到下一個用戶端定義之前的 *parameter* (如果已指定) 與媒體規格均特定於此用戶端。 *parameter* 與 *media* 定義必須縮排幾個空白或 tab 字元。

b. (可選擇) 編寫 *parameter* 欄位。

參數行以 *keyword* = *value* (數值) 對表示。您可以使用 *parameter* 欄位指定網路區段大小。 *net_block_size* (網路區段大小) 參數將指定此用戶端通訊端所用的網路區段大小 (以 KB 為單位)。此參數的格式如下所示：

```
net_blk_size=size
```

對於大小，請指定 $4 \leq size \leq 64$ 範圍內的整數。預設值為 4，它指定 4096 個位元組。
parameter 行必須定義縮排幾個空白或 tab 字元。

c. 編寫 `media` 與 `endmedia` 的關鍵字欄位。

伺服器配置檔案中需要 `media` 與 `endmedia` 關鍵字。這些關鍵字可定義用戶端使用的媒體歸檔磁碟區。這些媒體關聯指定如下：

程式碼範例 7-8 伺服器配置檔案中的媒體規格

```
media
    eq media-type (regex)
    [ eq media-type (regex) ]
    [ . . . ]
endmedia
```

`media` 與 `endmedia` 關鍵字將限定 Sun SAM-Remote 伺服器配置檔案的媒體定義區域。*eq media-type regex* 行為媒體定義行。`media` 必須定義縮排幾個空白或 tab 字元。*regex* 資料必須加上括號。

媒體類型規格的要素如下所示：

引數	定義
<i>eq</i>	程式庫的設備序號。 含混合媒體的網路連結式程式庫，可以具有多個 <i>eq media-type regex</i> 行。因此，請為每個媒體類型指定不同的 <i>eq media-type regex</i> 行。
<i>media-type</i>	以兩個字元表示的特定媒體類型，如 <code>lt</code> 。在 <code>mcf</code> 檔案中有效的一般媒體類型規格在 <i>media_type</i> 規格中無效。若需有關有效媒體類型的資訊，請參閱「 <code>mcf(4)</code> 線上手冊」。 您的網路連結式程式庫若涵括了多種媒體類型，請指定多個媒體定義行。例如：下列定義就是一個有效的媒體類型定義： <pre>media 100 lt (VSN1) 100 lt (VSN2) endmedia</pre>
<i>regex</i>	檔案所要歸檔的卡匣磁碟區序列名稱 (VSN)。每個 VSN 指定必須表示為延伸式常規表示式，且 VSN 必須加上括號。若需有關延伸正規式的資訊，請參閱「 <code>egrep(1)</code> 線上手冊」。 若需有關正規式的資訊，請參閱「 <code>regcomp(3C)</code> 線上手冊」。

備註 — 請禁止多個用戶端使用相同的實體媒體卡匣。此外，如果 Sun SAM-Remote 伺服器在 Sun SAM-Remote 環境以外具有自己的檔案系統，建議此用戶端與伺服器不要使用同一個卡匣。

▼ 啓用歸檔

下列步驟將啓用歸檔並完成配置程序。

1. 確認用戶端上的 `archiver.cmd` 檔案。

依您的配置而定，您可能需要執行下列作業：

- 確定伺服器配置檔案中定義的 VSN 已指派給 `archiver.cmd` 檔案中的正確歸檔組。
- 如果下列指令適用於要歸檔至與 Sun SAM-Remote 伺服器連接的程式庫之歸檔組，請從 Sun SAM-Remote 用戶端上的 `archiver.cmd` 檔案中移除這些指令：
 - `-tapenonstop`
 - `-offline_copy direct`

2. 輸入 `samd(1M)` 指令及其 `start` 選項，以啟動伺服器和用戶端上的 Sun StorageTek SAM 程序。

爲確保系統讀取伺服器與用戶端上的新配置檔案，您必須啓動或重新啓動 Sun StorageTek SAM 軟體。

在用戶端與伺服器上輸入下列指令：

```
server# samd start
```

如需有關啓動和重新啓動 Sun StorageTek SAM 的更完整指示，請參閱「Sun StorageTek Storage Archive Manager 安裝與升級指南」。

3. 在伺服器與用戶端上呼叫 `samu(1M)`。

此步驟的目的是確認主機之間的連線。使用 `samu(1M)` 公用程式的 `s` 與 `R` 畫面，顯示 Sun SAM-Remote 連線狀態。如需有關 `samu(1M)` 的更多資訊，請參閱「`samu(1M)` 線上手冊」或「Sun StorageTek QFS 檔案系統配置與管理指南」。

程式碼範例 7-9 顯示的是 Sun SAM-Remote 用戶端 portland 上的 samu(1M) 狀態 s 畫面。請注意裝置類型 sc，其表示 Sun SAM-Remote 用戶端。該行下面的訊息表示已建立伺服器 chicago 之間的連線。

程式碼範例 7-9 用戶端 samu(1M) s 畫面

```

Device status          samu    4.0.5 Wed May 02 14:44:44
License: License never expires.
ty    eq state  device_name          fs status  pos
ms    1 on     samfs1              1 m-----
md    10 on    /dev/dsk/c1t1d0s0        1 -----
md    12 on    /dev/dsk/c1t2d0s0        1 -----
s9    35 on    /dev/samst/c0t5u0        35 m-----r
      move complete
lt    36 on    /dev/rmt/0cbn            35 -----p
      empty
lt    37 on    /dev/rmt/1cbn            35 -----p
      empty
lt    38 on    /dev/rmt/2cbn            35 --l-----r
      idle
lt    39 on    /dev/rmt/3cbn            35 --l-----r
      idle
sc    200 on   /etc/opt/SUNWsamfs/rmt200  200 -----r
      server chicago connected
rd    201 on   /dev/samrd/rd0           200 -----r
rd    202 on   /dev/samrd/rd1           200 -----r
hy    203 on   historian                 203 -----

```

程式碼範例 7-10 顯示的是 Sun SAM-Remote 伺服器 chicago 上的 samu(1M) 狀態 s 畫面。請注意裝置類型 ss，其表示 Sun SAM-Remote 伺服器。此畫面指明此系統為 Sun SAM-Remote 伺服器。

程式碼範例 7-10 chicago 上的伺服器 samu(1M) s 畫面

```

Device status          samu    4.0.5 Tue Apr 24 14:49:43
License: License never expires.
ty    eq state  device_name          fs status  pos
ms    1 on     samfs1              1 m-----
md    11 on    /dev/dsk/c2t6d0s0        1 -----
md    12 on    /dev/dsk/c2t6d0s1        1 -----
ss    50 on    /etc/opt/SUNWsamfs/rmt200  50 -----r
sl    100 on   /dev/samst/c0t3u0        100 m-----r
at    101 on   /dev/rmt/0cbn            100 -----p
      initializing
at    102 on   /dev/rmt/1cbn            100 -----p
      initializing
hy    103 on   historian                 103 -----

```

程式碼範例 7-11 顯示的是 Sun SAM-Remote 伺服器 `chicago` 上的 `samu(1M)` Sun SAM-Remote R 畫面。

程式碼範例 7-11 `chicago` 上的伺服器 `samu(1M)` R 畫面

```
Remote server eq: 50                addr: 00001ca0 4.0.5 Wed May 02
14:55:37
message:

Client: portland
client index - 0
network block size - 4096
max file size - 0                    flags - c0000000
min file size - 8
```

如果您具有多個 Sun SAM-Remote 用戶端，則可按下 `CONTROL-f` 按鍵組合以捲動用戶端。

在程式碼範例 7-11 中，連接用戶端的命名為 `portland`。`client index` (用戶端索引) 欄位表示此用戶端是此伺服器常駐程式可能定義的 0 至 9 個用戶端中的第 0 個用戶端。所列的最大檔案大小、最小檔案大小及網路區段大小均以位元組為單位。旗標會指出連線的狀態，如表 7-1 所示：

表 7-1 `samu(1M)` R 畫面旗標

旗標	意義
0x00000000	無連線。
0xc0000000	已建立連線。

4. 在伺服器上使用 `samu(1M)` 公用程式的 `v` 畫面，以確定能夠在用戶端上使用目錄。

您應可檢視每個用戶端所提供的 Sun SAM-Remote 目錄檔。透過 `samu(1M)` 輸入下列指令：

```
:v eq
```

對於 `eq`，請依據 `mcf` 檔案中所定義，指定 Sun SAM-Remote 用戶端常駐程式的設備序號。

程式碼範例 7-12 會顯示 chicago 的 samu(1M) 畫面，指出 portland 所能夠存取的磁碟區。

程式碼範例 7-12 從 chicago 檢視時可用的磁碟區

```
Robot VSN catalog by slot : eq 200 samu 4.0.5 Wed May 02 15:24:13
count 32
slot      access      time  count use flags      ty vsn
  1      2003/01/02  10:40    0   0% -il-o-b-R-U-  at 000032
  2      2003/01/02  11:41    0   0% -il-o-b-R---  at 000034
  3      2003/01/02  12:42  170  91% -il-o-b-----  at 000035
  4      2003/01/02  13:43   20   7% -il-o-b-----  at 000037
  5      2003/01/02  14:44    0   0% -il-o-b-----  at 000038
  6      2003/01/02  13:41    0   0% -il-o-b-----  at 000031
```

5. 在每一個用戶端上輸入 archiver(1M) 指令及其 -A 選項。

在此步驟中，您將確認檔案是否從用戶端歸檔至伺服器。此指令可讓您從歸檔器寫入清單，包括伺服器上的 VSN。有關此指令的資訊，請參閱「archiver(1M) 線上手冊」。

檔案若不進行歸檔，請參閱「Sun StorageTek Storage Archive Manager 疑難排解指南」。

使用 Sun SAM-Remote 軟體進行回收

本節包含有關使用 Sun SAM-Remote 進行回收的資訊。Sun Microsystems 建議您只在此處所述之特定情況下，才在 Sun SAM-Remote 環境中執行回收。

由於回收程序涉及釋放卡匣上的空間以取得更多資料，因此，如果回收程序配置不正確，回收器可能會損毀歸檔匣上的重要資料。

備註 – Sun StorageTek SAM 軟體不會強制執行這些限制。

若要避免資料遺失，請務必注意下列限制：

- 在 Sun SAM-Remote 環境下使用回收器之前，必須完全瞭解回收器的每個步驟。以錯誤的順序或在錯誤的系統上執行指令，可能會導致資料遺失無法復原。在執行任何指令之前，請確定已分析指令的動作，例如：tplabel(1M)，這會刪除 Sun SAM-Remote 用戶端或 Sun SAM-Remote 伺服器上的資料。
- Sun SAM-Remote 伺服器與 Sun SAM-Remote 用戶端二者的回收活動不可重疊。否則，可能會導致意外重新標記卡匣與資料遺失無法復原。
- 不可回收內含可移除式媒體檔案的卡匣。

- 在 Sun SAM-Remote 用戶端與伺服器環境中，用戶端與伺服器無法辨識彼此的檔案系統、資料檔案及 inode 檔案。因此，伺服器與用戶端必須各自擁有專屬的卡匣組。此外亦不應使用對方的卡匣。

您可以在 Sun SAM-Remote 伺服器的 `/etc/opt/SUNWsamfs/recycler.cmd` 檔案中建立 `no_recycle` 清單，來防止意外回收 Sun SAM-Remote 用戶端使用的 VSN。但在對 `no_recycle` 清單中之磁碟區使用 `chmed(1M)` 指令的 `+c` 選項時，請務必小心。使用此指令設定磁碟區上的回收旗標 (`+c`) 時，該動作會覆寫 `/etc/opt/SUNWsamfs/recycler.cmd` 檔案中的 `no_recycle` 清單。

- 請勿嘗試在同一天回收 Sun SAM-Remote 伺服器與 Sun SAM-Remote 用戶端上的磁碟區。

唯有符合下列條件時，才能在 Sun SAM-Remote 環境下進行回收：

- 系統中的每個 VSN 只能由一個用戶端系統或伺服器使用。任何 VSN 上的檔案不能由多個系統使用。
- 除了包含該用戶端歸檔影像的 VSN 之外，Sun SAM-Remote 用戶端沒有任何 VSN 的目錄項目。伺服器配置檔案之媒體定義行中的 *regex (eq media-type regex 行)* 必須和用戶端目錄中所指定的磁碟區相同。此外，用戶端目錄中的 *regex* 規格無法指定相同的磁碟區。
- 根據歸檔組進行歸檔。使用 Sun SAM-Remote 時，必須依歸檔組 (而非程式庫) 執行回收。

以下幾節將說明使用 Sun SAM-Remote 用戶端與伺服器啓用回收的兩種方法。方法如下所示：

- [第 165 頁的「於 Sun SAM-Remote 環境中進行回收 – 方法 1」](#)
- [第 188 頁的「於 Sun SAM-Remote 環境中進行回收 – 方法 2」](#)

於 Sun SAM-Remote 環境中進行回收 – 方法 1

本節中的程序將說明在具有伺服器 `sky` 與用戶端 `zeke` 之 Sun SAM-Remote 環境中啓用回收的方法之一。



注意 – 您必須完成本程序所有步驟並測試配置，以驗證回收是否正確執行之後，才可在 Sun SAM-Remote 環境中使用回收器。

伺服器 sky 的配置檔案

伺服器在 mcf 檔案及其伺服器配置檔案中必須具有 Sun SAM-Remote 配置資訊。下列代碼範例顯示了這些檔案。

程式碼範例 7-13 顯示的是伺服器 sky 上的 mcf 檔案。

程式碼範例 7-13 伺服器 sky 上的 mcf 檔案

```
# This is the mcf file for the server (sky).
# The server parameters file (rmt1000) points
#   back to the correct automated library's equipment number
#   (70) for the ADIC Scalar 1000.
#
samfs1          100   ma   samfs1   on
/dev/dsk/c0t0d0s5 110   mm   samfs1   on   /dev/rdisk/c0t0d0s5
/dev/dsk/c3t2d0s3 120   mr   samfs1   on   /dev/rdisk/c3t2d0s3
/dev/dsk/c3t2d0s4 121   mr   samfs1   on   /dev/rdisk/c3t2d0s4
samfs2          139   ma   samfs2   on
/dev/dsk/c3t4d0s3 140   mm   samfs2   on   /dev/rdisk/c3t4d0s3
/dev/dsk/c3t4d0s4 141   mr   samfs2   on   /dev/rdisk/c3t4d0s4
# ADIC Scalar 1000
/dev/samst/c0t0u0 70   rb   adic1   -   /var/opt/SUNWsamfs/catalog/adic1
/dev/rmt/0bn     71   at   adic1   on
/dev/rmt/1bn     72   at   adic1   on
/dev/rmt/2bn     73   at   adic1   on
/dev/rmt/3bn     74   at   adic1   on
/dev/rmt/4bn     75   at   adic1   on
/dev/rmt/5bn     76   at   adic1   on
/dev/rmt/11bn    77   at   adic1   on
/dev/rmt/10bn    78   at   adic1   on
/dev/rmt/9bn     79   at   adic1   on
/dev/rmt/8bn     80   at   adic1   on
/dev/rmt/7bn     81   at   adic1   on
/dev/rmt/6bn     82   at   adic1   on
# Define Sun SAM-Remote server skyrs
/etc/opt/SUNWsamfs/rmt1000 1000 ss skyrs on
```

程式碼範例 7-14 顯示的是伺服器 sky 上的伺服器配置檔案。

程式碼範例 7-14 伺服器 sky 上的伺服器配置檔案

```
# Server configuration file /etc/opt/SUNWsamfs/rmt1000 on sky.
# The eq of the automated library MUST match the eq of the
#   automated library that you want to use in the mcf file.
zeke
  media
  70 at 00002[0-9]
endmedia
```

用戶端 zeke 的配置檔案

用戶端在 mcf 檔案及其用戶端配置檔案中必須具有 Sun SAM-Remote 配置資訊。下列代碼範例顯示了這些檔案。

程式碼範例 7-15 顯示的是用戶端 zeke 上的 mcf 檔案。

程式碼範例 7-15 用戶端 zeke 上的 mcf 檔案

```
# mcf file for client (zeke)
#
samfs1          10  ms  samfs1  on
/dev/dsk/c1t3d0s0  11  md  samfs1  on  /dev/rdisk/c1t3d0s0
/dev/dsk/c1t3d0s1  12  md  samfs1  on  /dev/rdisk/c1t3d0s1
/dev/dsk/c1t3d0s3  13  md  samfs1  on  /dev/rdisk/c1t3d0s3

# Define a StorageTek L20 with 1 drive and 20 slots (including cap)
/dev/samst/c0t2u0  50  rb  stk_l20  on  /var/opt/SUNWsamfs/catalog/L20_cat
/dev/rmt/0hbn     51  lt  stk_l20  on

# Define zeke as a Sun SAM-Remote client using sky as the server
/etc/opt/SUNWsamfs/sky 200  sc  skyrs   on  /var/opt/SUNWsamfs/catalog/sky_cat
/dev/samrd/rd0      201  rd  skyrs   on
/dev/samrd/rd1      202  rd  skyrs   on
/dev/samrd/rd2      203  rd  skyrs   on
/dev/samrd/rd3      204  rd  skyrs   on
```

程式碼範例 7-16 顯示用戶端 zeke 上的用戶端配置檔案。

程式碼範例 7-16 用戶端 zeke 上的用戶端配置檔案

```
# cat /etc/opt/SUNWsamfs/sky
# File /etc/opt/SUNWsamfs/sky on Sun SAM-Remote client zeke:
sky
```

▼ 配置回收 — 方法 1

配置回收程序時，必須對歸檔與回收進行測試。由於需要測試，因此本程序可能需要一兩天才能完成，視歸檔與回收檔案有多頻繁而定。

在啟動程序之前，請先閱讀第 6 章中有關回收器的資訊。在 Sun SAM-Remote 環境下使用回收器必須完全瞭解回收程序中的步驟。如果您不熟悉回收程序，請立即花些時間加以瞭解。

備註 — 請勿在伺服器上使用 `chmed(1M)` 指令來為用戶端 VSN 設定回收旗標 (+c)。該動作會覆寫伺服器 `/etc/opt/SUNWsamfs/recycler.cmd` 檔案中的 `no_recycle` 清單。

1. 確定 Sun SAM-Remote 用戶端與伺服器已正確配置且會進行歸檔。

如需有關配置與驗證 Sun SAM-Remote 環境的更多資訊，請參閱第 152 頁的「[配置 Sun SAM-Remote 軟體](#)」，其中包含有關配置 Sun SAM-Remote 用戶端與伺服器的詳細資訊。該程序包含確定進行歸檔的步驟。

2. 編輯用戶端系統上的 archiver.cmd 檔案，然後新增回收指令。

在本範例中，根據歸檔組 (而不是程式庫) 進行回收。指定由歸檔組進行回收的指令必須出現在 archiver.cmd 檔案中。

[程式碼範例 7-17](#) 顯示的是用戶端 zeke 上的 archiver.cmd 檔案。此檔案已編輯以與回收器進行通訊。

程式碼範例 7-17 用戶端 zeke 上的 archiver.cmd 檔案

```
# This is file /etc/opt/SUNWsamfs/archiver.cmd
# on Sun SAM-Remote client zeke.
#
# wait
logfile = /var/opt/SUNWsamfs/archiver/archiver.log
trace = /var/opt/SUNWsamfs/trace/archiver all
interval = 1m
no_archive tmp
no_archive .
archmax = lt 2G
archmax = at 5G
drives = skyr5 4 # use up to four drives for remote archiving.
fs = samfs1
    1 4h
archiveset testdir0
    1 1m
    2 1m
defaultset .
    1 1m
    2 1m

params
# Start with mingain high to reduce workload.
# If you need more recycling, reduce mingain.
# If too much recycling, increase High-Water Mark.
archiveset.1 -recycle_hwm 60 -recycle_mingain 90 -recycle_vsncount 1
archiveset.1 -recycle_ignore
defaultset.1 -recycle_hwm 60 -recycle_mingain 90 -recycle_vsncount 1
defaultset.1 -recycle_ignore
```


程式碼範例 7-17 用戶端 zeke 上的 archiver.cmd 檔案 (續)

```
# Remote directives.
# Use up to three drives per archive set.
# Load will split to two drives at 100m, to three drives at 150m.
archiveset.2 -drives 3 -drivemin 50m
defaultset.2 -drives 3 -drivemin 50m

# Remote directives.
# Start with mingain high to reduce workload.
# If you need more recycling, reduce mingain.
# If too much recycling, increase High-Water Mark.
archiveset.2 -recycle_hwm 60 -recycle_mingain 90 -recycle_vsncount 1
archiveset.2 -recycle_ignore
defaultset.2 -recycle_hwm 60 -recycle_mingain 90 -recycle_vsncount 1
defaultset.2 -recycle_ignore
endparams

vsns
samfs1.1      lt 000173      # local copy.
archiveset.1  lt ^CEL       # local copy.
archiveset.2  at 00002[0-4] # remote copy, sky ait-2
                                     # tapes 20 through 24.
defaultset.1  lt ^CSM       # local copy.
defaultset.2  at 00002[5-9] # remote copy, sky ait-2
                                     # tapes 25 through 29.
endvsns
```

程式碼範例 7-17 中所示的指令會執行下列作業：

- `-recycle_hwm` 指令用於設定程式庫之歸檔集的高標記。VSN 的使用率超過此百分比時，會開始回收歸檔組。
- `-recycle_ignore` 指令只是暫時插入。此指令可防止在配置與測試環境之前進行回收。您可以在稍後的步驟中移除此指令。
- 將 `-recycle_mingain` 指令設定為高，以限制必須重新取得空間之工作的數量，藉此確保效能。
- `-recycle_vsncount 1` 指令指定回收器每次只清空一個 VSN。清空第一個 VSN 時，會選擇另一個要開始清空的 VSN。無論何時，皆只會對佇列中的一個 VSN 進行重新標記，並只會清空佇列中的一個 VSN。如此一來，回收作業便不會造成系統負擔過重。

3. 編輯用戶端上的 `recycler.cmd` 檔案，並指定記錄檔以接收回收記錄輸出。

程式碼範例 7-18 顯示的是用戶端 `zeke` 上經過編輯，以指定回收器記錄檔的 `recycler.cmd` 檔案。

程式碼範例 7-18 用戶端 `zeke` 上的 `recycler.cmd` 檔案

```
#
# This is the /etc/opt/SUNWsamfs/recycler.cmd file
# on client zeke.
#
logfile = /var/opt/SUNWsamfs/log/recycler
```

4. 確認已寫入伺服器上的 `archiver.cmd` 檔案，以按歸檔組指定回收。

使用 Sun SAM-Remote 時，您必須指定根據歸檔組 (而不是程式庫) 進行回收。指定由歸檔組進行回收的指令必須出現在 `archiver.cmd` 檔案中。

程式碼範例 7-19 顯示 `sky` 伺服器上的 `archiver.cmd` 檔案。此檔案指定按歸檔組進行歸檔。

程式碼範例 7-19 伺服器 `sky` 上的 `archiver.cmd` 檔案

```
# This is the archiver.cmd for the server (sky).
#
# Number of drives: 10
# Number of Mounted Filesystems: 1
# Number of Tests per Filesystem: 1
# Number of Archive Copies per Test: 2
#wait
#trace = /var/opt/SUNWsamfs/trace/archiver all

logfile = /var/opt/SUNWsamfs/log/archiver
interval = 1m
no_archive .
archmax = at 5G
drives = adicl 6
fs = samfs1
    1 4h
testset testdir0
    1 1m
    2 1m
allsam1 .
    1 1m
    2 1m
params
```

程式碼範例 7-19 伺服器 sky 上的 archiver.cmd 檔案 (續)

```
allsam1.1 -drives 4 -drivemin 50m
allsam1.1 -recycle_hwm 60 -recycle_mingain 90 -recycle_vsncount 1
allsam1.1 -recycle_ignore
allsam1.2 -drives 4 -drivemin 50m
allsam1.2 -recycle_hwm 60 -recycle_mingain 90 -recycle_vsncount 1
allsam1.2 -recycle_ignore
testset.1 -drives 4 -drivemin 50m
testset.1 -recycle_hwm 60 -recycle_mingain 90 -recycle_vsncount 1
testset.1 -recycle_ignore
testset.2 -drives 4 -drivemin 50m
testset.2 -recycle_hwm 60 -recycle_mingain 90 -recycle_vsncount 1
testset.2 -recycle_ignore
endparams
vsns
samfs1.1 at 000000
allsam1.1 at 00000[1-5] # vsns 1 through 5.
allsam1.2 at 00000[6-9] # vsns 6 through 9.
testset.1 at 00001[0,4] # vsns 10 and 14.
testset.2 at 00001[5,9] # vsns 15 and 19.
endvsns
```

5. 編輯伺服器上的 recycler.cmd 檔案。

修改此檔案，以指定下列項目：

- 從回收器接收執行結果的回收器記錄檔。
- Sun SAM-Remote 用戶端 VSN 的 no_recycle 指令。Sun SAM-Remote 用戶端配置為將其備份 2 歸檔備份寫入 Sun SAM-Remote 伺服器程式庫中的卡匣。要防止用於歸檔的 Sun SAM-Remote 用戶端使用的 VSN 被 Sun SAM-Remote 伺服器回收，必須要有 no_recycle 指令。

程式碼範例 7-20 顯示的是伺服器 sky 上經過編輯，以指定回收器記錄檔的 recycler.cmd 檔案。

程式碼範例 7-20 伺服器 sky 上的 recycler.cmd 檔案

```
#
# This is the /etc/opt/SUNWsamfs/recycler.cmd file
# on Sun SAM-Remote server sky.
#
logfile = /var/opt/SUNWsamfs/recycler/recycler.log
adic1 -ignore
no_recycle at 00002[0-9] # Prevents VSNs assigned to zeke from
                        # being recycled.
```

6. 使用 sam-recycler(1M) 指令在 Sun SAM-Remote 用戶端上測試回收器。

在 Sun SAM-Remote 用戶端系統上執行回收器。這項測試可查看回收器是否已正確地確認配置檔案中指定的裝置與 VSN。

例如，您可以使用下列指令執行回收器的初始測試：

```
zeke# sam-recycler -dvx
```

此測試十分重要，因為回收器若是偵測到其執行所在之系統上的系統目錄 (包括記錄器目錄) 中，沒有所列之特定 VSN 的歸檔影像，recycler.sh 程序檔即可能會要求重新標記卡匣。標記卡匣會損毀卡匣上的所有資料。Sun SAM-Remote 用戶端與 Sun StorageTek SAM 伺服器之間沒有通訊方式可通知對方有關歸檔副本的存在。所有此類資訊都是由本機 Sun StorageTek SAM 檔案系統提供。

回收器將執行並將其活動記錄至回收器記錄檔。回收器記錄檔將在 recycler.cmd 檔案中定義。若需有關 sam-recycler(1M) 指令的更多資訊，請參閱「sam-recycler(1M) 線上手冊」。

7. 檢查回收器記錄檔。

您將尋找下列訊息：

```
Recycling is ignored on this archive set.
```

程式碼範例 7-21 顯示的是記錄檔範例。

程式碼範例 7-21 用戶端 zeke 上的回收器記錄檔

```
# recycler.log from client zeke.
===== Recycler begins at Mon Jun  4 09:49:41 2001 =====
Initial 7 catalogs:
0  Family: stk_l20                Path: /var/opt/SUNWsamfs/catalog/L20_cat
   Vendor: STK                    Product: L20
   SLOT          ty      capacity      space vsn
   0              lt      33.0G        33.0G 000173
   1              lt      32.8G        44.1M CEL170
   2              lt      33.0G        33.0G CEL139
   4              lt      32.8G        16.8G CFC504
   5              lt      33.0G        33.0G CFC503
   6              lt      32.9G         0      CSM689
   7              lt      32.9G        19.6G CSM690
   8              lt      33.0G        33.0G CSM691
   9              lt      33.0G        33.0G CSM692
  10             lt      10.0G        10.0G CLN018
  11             lt      33.0G        33.0G 000766
Total Capacity:  339.2G bytes, Total Space Available: 244.3G bytes
Volume utilization 27%, high 95% VSN_min 50%
Recycling is ignored on this robot.
```

```

1 Family: skyrs                      Path: /var/opt/SUNWsamfs/catalog/sky_cat
  Vendor: (NULL)                     Product: (NULL)
  SLOT          ty    capacity        space vsn
    0            at    48.5G          23.3G 000020
    1            at    23.8G          23.8G 000021
    2            at    48.5G          48.5G 000022
    3            at    48.5G          48.5G 000023
    4            at    48.5G          48.5G 000024
    5            at    48.5G          2.6G 000025
    6            at    48.5G          361.4k 000026
    7            at    48.5G          48.5G 000027
    8            at    48.5G          48.5G 000028
    9            at    48.5G          0      000029
  Total Capacity: 460.8G bytes, Total Space Available: 292.5G bytes
  Volume utilization 36%, high 95% VSN_min 50%
  Recycling is ignored on this robot.

2 Family: hy                          Path: /var/opt/SUNWsamfs/catalog/historian
  Vendor: Sun SAM-FS                  Product: Historian
  SLOT          ty    capacity        space vsn
  (no VSNs in this media changer)
  Total Capacity: 0 bytes, Total Space Available: 0 bytes
  Volume utilization 0%, high 95% VSN_min 50%
  Recycling is ignored on this robot.

3 Family: defaultset.1                Path: /etc/opt/SUNWsamfs/archiver.cmd
  Vendor: Sun SAM-FS                  Product: Archive set
  SLOT          ty    capacity        space vsn
    0            lt    33.0G          33.0G 000766
    1            lt    33.0G          33.0G 000173
    2            lt    32.9G          0      CSM689
    3            lt    32.9G          19.6G CSM690
    4            lt    33.0G          33.0G CSM691
    5            lt    33.0G          33.0G CSM692
  Total Capacity: 197.6G bytes, Total Space Available: 151.5G bytes
  Volume utilization 23%, high 60% VSN_min 90%
  Recycling is ignored on this archive set.

```

```

4 Family: defaultset.2          Path: /etc/opt/SUNWsamfs/archiver.cmd
  Vendor: Sun SAM-FS           Product: Archive set
  SLOT          ty      capacity      space vsn
    0            lt      32.9G         0     CSM689
    1            at      48.5G         23.3G 000020
    2            at      23.8G         23.8G 000021
    3            at      48.5G         2.6G  000025
    4            at      48.5G         361.4k 000026
    5            at      48.5G         48.5G 000027
    6            at      48.5G         48.5G 000028
    7            at      48.5G         0     000029
  Total Capacity: 348.0G bytes, Total Space Available: 146.8G bytes
  Volume utilization 57%, high 60% VSN_min 90%
  Recycling is ignored on this archive set.

5 Family: archiveset.1         Path: /etc/opt/SUNWsamfs/archiver.cmd
  Vendor: Sun SAM-FS           Product: Archive set
  SLOT          ty      capacity      space vsn
    0            lt      32.8G         44.1M CEL170
    1            lt      32.8G         16.8G CFC504
    2            lt      33.0G         33.0G CFC503
  Total Capacity: 98.6G bytes, Total Space Available: 49.8G bytes
  Volume utilization 49%, high 60% VSN_min 90%
  Recycling is ignored on this archive set.

6 Family: archiveset.2         Path: /etc/opt/SUNWsamfs/archiver.cmd
  Vendor: Sun SAM-FS           Product: Archive set
  SLOT          ty      capacity      space vsn
    0            at      48.5G         23.3G 000020
    1            at      23.8G         23.8G 000021
    2            at      48.5G         48.5G 000022
    3            at      48.5G         48.5G 000023
    4            at      48.5G         48.5G 000024
  Total Capacity: 218.0G bytes, Total Space Available: 192.8G bytes
  Volume utilization 11%, high 60% VSN_min 90%
  Recycling is ignored on this archive set.

```

21 VSNs:

```

---Archives---      -----Percent-----      defaultset.1
-----Status-----      Count      Bytes      Use Obsolete Free      Library:Type:VSN
in multiple sets      0      0      0 100      0      stk_l20:lt:CSM689
partially full      111      2.8G      8 31      61      stk_l20:lt:CSM690
empty VSN      0      0      0 0      100      stk_l20:lt:000173
empty VSN      0      0      0 0      100      stk_l20:lt:CSM691
empty VSN      0      0      0 0      100      stk_l20:lt:CSM692
empty VSN      0      0      0 0      100      stk_l20:lt:000766

---Archives---      -----Percent-----      defaultset.2
-----Status-----      Count      Bytes      Use Obsolete Free      Library:Type:VSN
no-data VSN      0      0      0 100      0      skyrs:at:000029
no-data VSN      0      0      0 99      1      skyrs:at:000026
partially full      111      2.8G      6 88      6      skyrs:at:000025
empty VSN      0      0      0 0      100      skyrs:at:000028
empty VSN      0      0      0 0      100      skyrs:at:000027

---Archives---      -----Percent-----      archiveset.1
-----Status-----      Count      Bytes      Use Obsolete Free      Library:Type:VSN
no-data VSN      0      0      0 99      1      stk_l20:lt:CEL170
partially full      677      2.3G      8 40      52      stk_l20:lt:CFC504
empty VSN      0      0      0 0      100      stk_l20:lt:CFC503

---Archives---      -----Percent-----      archiveset.2
-----Status-----      Count      Bytes      Use Obsolete Free      Library:Type:VSN
in multiple sets      0      0      0 51      49      skyrs:at:000020
empty VSN      0      0      0 0      100      skyrs:at:000022
empty VSN      0      0      0 0      100      skyrs:at:000023
empty VSN      0      0      0 0      100      skyrs:at:000024
in multiple sets      0      0      0 0      100      skyrs:at:000021

---Archives---      -----Percent-----      stk_l20
-----Status-----      Count      Bytes      Use Obsolete Free      Library:Type:VSN
empty VSN      0      0      0 0      100      stk_l20:lt:CLN018
partially full      13      80.3k      0 0      100      stk_l20:lt:CEL139
Recycler finished.
===== Recycler ends at Mon Jun 4 09:49:53 2001 =====

```

8. 在 Sun SAM-Remote 伺服器上輸入 `sam-recycler(1M)` 指令，以確定回收器未回收任何保留供 Sun SAM-Remote 用戶端使用的 VSN。

例如：

```
sky# sam-recycler -dvx
```

上述指令將執行回收器並將其活動寫入回收器記錄檔。若需有關 sam-recycler(1M) 指令的更多資訊，請參閱「sam-recycler(1M) 線上手冊」。

程式碼範例 7-22 顯示範例回收器記錄檔。

程式碼範例 7-22

回收器記錄檔

```
# recycler.log file from server sky.
===== Recycler begins at Mon Jun  4 09:50:44 2001 =====
Initial 6 catalogs:
0  Family: adicl                Path: /var/opt/SUNWsamfs/catalog/adicl
   Vendor: ADIC                 Product: Scalar 1000
   SLOT          ty      capacity      space vsn
   0             at       1.3G          1.2G 000001
   1             at       1.3G          1.3G 000002
   2             at       1.3G          1.3G 000004
   3             at      48.5G           0    000010
   4             at      48.5G           0    000011
   5             at      48.5G          43.5G 000018
   6             at      48.5G           0    000019
   7             at      48.5G          23.3G 000020
   8             at      23.8G          23.8G 000021
   9             at      48.5G          48.5G 000022
  10            at      48.5G          48.5G 000023
  11            at      48.5G          48.5G 000024
  12            at      48.5G           2.6G 000025

  13            at      48.5G          361.4k 000026
  14            at      48.5G          48.5G 000027
  15            at      48.5G          48.5G 000028
  16            at      48.5G           0    000029
  17            at       1.3G          1.3G 000005
  18            at      48.5G          48.5G 000016
  19            at      23.8G          23.8G CLN001
  20            at      23.8G          23.8G CLN002
  21            at      23.8G          23.8G CLN004
  22            at      23.8G          23.8G CLN003
  23            at      48.5G          421.6M 000015
  24            at       1.3G          1.3G 000000
  25            at      48.5G           0    000013
```



```

26          at          1.3G          1.3G 000003
27          at          48.5G         43.6G 000007
28          at          48.5G         41.8G 000008
29          at          48.5G         46.9G 000006
30          at          48.5G         48.3G 000009
31          at          48.5G          0    000014
32          at          48.5G          0    000012
33          at          48.5G         40.1G 000017
Total Capacity: 1.2T bytes, Total Space Available: 708.7G bytes
Volume utilization 43%, high 95% VSN_min 50%
Recycling is ignored on this robot.

1 Family: hy                      Path: /var/opt/SUNWsamfs/catalog/historian
Vendor: Sun SAM-FS                Product: Historian
SLOT          ty    capacity      space vsn
  (no VSNs in this media changer)
Total Capacity: 0    bytes, Total Space Available: 0    bytes
Volume utilization 0%, high 95% VSN_min 50%
Recycling is ignored on this robot.

2 Family: testset.1              Path: /etc/opt/SUNWsamfs/archiver.cmd
Vendor: Sun SAM-FS                Product: Archive set
SLOT          ty    capacity      space vsn
  0            at    48.5G         0    000010
  1            at    48.5G         0    000014
Total Capacity: 97.1G bytes, Total Space Available: 0    bytes
Volume utilization 100%, high 60% VSN_min 90%: *** Needs recycling ***
Recycling is ignored on this archive set.

3 Family: testset.2              Path: /etc/opt/SUNWsamfs/archiver.cmd
Vendor: Sun SAM-FS                Product: Archive set
SLOT          ty    capacity      space vsn
  0            at    48.5G         0    000019
  1            at    48.5G         421.6M 000015
Total Capacity: 97.1G bytes, Total Space Available: 421.6M bytes
Volume utilization 99%, high 60% VSN_min 90%: *** Needs recycling ***
Recycling is ignored on this archive set.

```

```

4 Family: allsam1.1          Path: /etc/opt/SUNWsamfs/archiver.cmd
  Vendor: Sun SAM-FS        Product: Archive set
  SLOT      ty      capacity      space vsn
    0       at       1.3G          1.2G 000001
    1       at       1.3G          1.3G 000002
    2       at       1.3G          1.3G 000004
    3       at       1.3G          1.3G 000005
    4       at       1.3G          1.3G 000003
Total Capacity: 6.5G bytes, Total Space Available: 6.3G bytes
Volume utilization 3%, high 60% VSN_min 90%
Recycling is ignored on this archive set.

```

```

5 Family: allsam1.2          Path: /etc/opt/SUNWsamfs/archiver.cmd
  Vendor: Sun SAM-FS        Product: Archive set
  SLOT      ty      capacity      space vsn
    0       at      48.5G          43.6G 000007
    1       at      48.5G          41.8G 000008
    2       at      48.5G          46.9G 000006
    3       at      48.5G          48.3G 000009
Total Capacity: 194.2G bytes, Total Space Available: 180.6G bytes
Volume utilization 6%, high 60% VSN_min 90%
Recycling is ignored on this archive set.

```

```

Need to select candidate for media changer testset.1 to free up 39.8G bytes.
Quantity of data to move limited to (no limit) bytes and 1 VSNs.
Checking 000010. Need to free 39.8G, quantity limit: (no limit), VSN count: 1.
  VSN is in correct media changer... good.
  VSN is not already recycling... good.
  VSN has no request files... good.
  VSN has no 'archive -n' files...good.
  VSN was not specified as "no_recycle" in recycler.cmd file... good.
  VSN does not exceed VSN count limit... good.
  VSN does not exceed data quantity limit... good.
  VSN meets minimum gain requirement.
  Recycling is ignored on this media changer - VSN not marked for recycling.

```

```
Checking 000014. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN is in correct media changer... good.
  VSN is not already recycling... good.
  VSN has no request files... good.
  VSN has no 'archive -n' files...good.
  VSN was not specified as "no_recycle" in recycler.cmd file... good.
  VSN exceeds VSN count limit - skipped.
Checking 000019. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000015. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.

Checking 000001. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000003. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000004. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000005. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000002. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000008. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000007. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.

Checking 000006. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000009. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000011. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000029. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000013. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000012. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
```

```
Checking 000026. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000025. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000020. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000017. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000018. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking CLN003. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.

Checking 000021. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000022. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000027. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000028. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000023. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000024. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000016. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.

Checking CLN001. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking CLN002. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking CLN004. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000000. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
No candidate was found in this media changer.

Need to select candidate for media changer testset.2 to free up 38.8G bytes.
Quantity of data to move limited to (no limit) bytes and 1 VSNs.
Checking 000010. Need to free 38.8G, quantity limit: (no limit), VSN count: 1.
VSN not in correct media changer.
Checking 000014. Need to free 38.8G, quantity limit: (no limit), VSN count: 1.
VSN not in correct media changer.
```

```
Checking 000019. Need to free 38.8G, quantity limit: (no limit), VSN count: 1.
  VSN is in correct media changer... good.
  VSN is not already recycling... good.
  VSN has no request files... good.
  VSN has no 'archive -n' files...good.
  VSN was not specified as "no_recycle" in recycler.cmd file... good.
  VSN does not exceed VSN count limit... good.
  VSN does not exceed data quantity limit... good.
  VSN meets minimum gain requirement.
  Recycling is ignored on this media changer - VSN not marked for recycling.

Checking 000015. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN is in correct media changer... good.
  VSN is not already recycling... good.
  VSN has no request files... good.
  VSN has no 'archive -n' files...good.
  VSN was not specified as "no_recycle" in recycler.cmd file... good.
  VSN exceeds VSN count limit - skipped.

Checking 000001. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000003. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000004. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000005. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000002. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.

Checking 000008. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000007. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000006. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000009. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000011. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
Checking 000029. Need to free 0E, quantity limit: (no limit), VSN count: 0.
  VSN not in correct media changer.
```

```
Checking 000013. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000012. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000026. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000025. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000020. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000017. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.

Checking 000018. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking CLN003. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000021. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000022. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000027. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000028. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000023. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.

Checking 000024. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000016. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking CLN001. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking CLN002. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking CLN004. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000000. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
No candidate was found in this media changer.
34 VSNs:
```

	---Archives---	-----Percent-----	testset.1			
-----Status-----	Count	Bytes	Use	Obsolete	Free	Library:Type:VSN
no-data VSN	0	0	0	100	0	adic1:at:000010
no-data VSN	0	0	0	100	0	adic1:at:000014
	---Archives---	-----Percent-----	testset.2			
-----Status-----	Count	Bytes	Use	Obsolete	Free	Library:Type:VSN
no-data VSN	0	0	0	100	0	adic1:at:000019
partially full	677	2.3G	5	93	2	adic1:at:000015
	---Archives---	-----Percent-----	allsam1.1			
-----Status-----	Count	Bytes	Use	Obsolete	Free	Library:Type:VSN
partially full	97	173.8M	1	9	90	adic1:at:000001
no-data VSN	0	0	0	2	98	adic1:at:000003
no-data VSN	0	0	0	2	98	adic1:at:000004
empty VSN	0	0	0	0	100	adic1:at:000005
empty VSN	0	0	0	0	100	adic1:at:000002
	---Archives---	-----Percent-----	allsam1.2			
-----Status-----	Count	Bytes	Use	Obsolete	Free	Library:Type:VSN
no-data VSN	0	0	0	13	87	adic1:at:000008
partially full	98	1.6G	3	7	90	adic1:at:000007
no-data VSN	0	0	0	3	97	adic1:at:000006
empty VSN	0	0	0	0	100	adic1:at:000009
	---Archives---	-----Percent-----	adic1			
-----Status-----	Count	Bytes	Use	Obsolete	Free	Library:Type:VSN
no-data VSN	0	0	0	100	0	adic1:at:000011
no_recycle VSN	0	0	0	100	0	adic1:at:000029
no-data VSN	0	0	0	100	0	adic1:at:000013
no-data VSN	0	0	0	100	0	adic1:at:000012
no_recycle VSN	0	0	0	99	1	adic1:at:000026
no_recycle VSN	0	0	0	94	6	adic1:at:000025
no_recycle VSN	0	0	0	51	49	adic1:at:000020
no-data VSN	0	0	0	17	83	adic1:at:000017
no-data VSN	0	0	0	10	90	adic1:at:000018
empty VSN	0	0	0	0	100	adic1:at:CLN003

```

no_recycle VSN      0      0      0      0      100  adic1:at:000021
no_recycle VSN      0      0      0      0      100  adic1:at:000022
no_recycle VSN      0      0      0      0      100  adic1:at:000027
no_recycle VSN      0      0      0      0      100  adic1:at:000028
no_recycle VSN      0      0      0      0      100  adic1:at:000023
no_recycle VSN      0      0      0      0      100  adic1:at:000024
empty VSN           0      0      0      0      100  adic1:at:000016
empty VSN           0      0      0      0      100  adic1:at:CLN001
empty VSN           0      0      0      0      100  adic1:at:CLN002
empty VSN           0      0      0      0      100  adic1:at:CLN004
partially full      12     88.3k  0      0      100  adic1:at:000000

```

Recycler finished.

===== Recycler ends at Mon Jun 4 09:51:05 2001 =====

9. 分析伺服器與用戶端的 recycler.log 檔案，以選擇要進行回收的候選 VSN。

接近 recycler.log 檔案結尾的是 Status 欄。用戶端記錄檔中具有下列狀態項目類型的 VSN，即是要進行回收的候選者：

- no-data VSN。要回收 no-data VSN，請參閱第 184 頁的「回收 no-data VSN」。
- partially full。要回收 partially full VSN，請參閱第 186 頁的「回收 partially full VSN」。

在伺服器記錄檔中，最佳的回收候選者是 Count、Bytes 及 Use 欄中數值為 0 的項目。

▼ 回收 no-data VSN

no-data VSN 是最易於回收的 VSN。這些 VSN 的 Count、Bytes 及 Use 欄位值均為 0 (零)。

1. 檢查用戶端中的 recycler.log 檔案，查看是否有任何 no-data VSN。

由於用戶端 zeke 中的 VSN 000029 與 000026 是 no-data VSN，因此可考慮對其進行回收，如程式碼範例 7-23 中所示。

程式碼範例 7-23 用戶端 zeke 上的 recycler.log 檔案

```

# From the client zeke recycler.log file:
          ---Archives---  -----Percent-----  defaultset.2
-----Status-----  Count  Bytes  Use Obsolete Free  Library:Type:VSN
no-data VSN           0      0      0   100      0  skyrs:at:000029
no-data VSN           0      0      0    99      1  skyrs:at:000026
partially full       111    2.8G    6    88      6  skyrs:at:000025
empty VSN             0      0      0     0     100  skyrs:at:000028
empty VSN             0      0      0     0     100  skyrs:at:000027

```


2. 檢查伺服器中的 `recycler.log` 檔案，並判斷伺服器的回收器記錄檔案中是否同樣顯示了您從上一步中選擇的 **VSN**。

您想要驗證伺服器上是否沒有任何使用中的資料歸檔到這些 **VSN** 上。

程式碼範例 7-24 顯示的是伺服器之 `recycler.log` 檔案中的 `no_recycle` **VSN** 資料。對 **VSN** 000029 與 000026 而言，伺服器之 `recycler.log` 檔案與用戶端之 `recycler.log` 檔案所記錄的資料相同。

程式碼範例 7-24 伺服器 sky 上的 `recycler.log` 檔案

```
# From the Server log file:
      ---Archives---      -----Percent-----      adic1
-----Status-----      Count      Bytes      Use Obsolete Free      Library:Type:VSN
no-data VSN                0          0          0  100      0      adic1:at:000011
no_recycle VSN             0          0          0  100      0      adic1:at:000029
no-data VSN                0          0          0  100      0      adic1:at:000013
no-data VSN                0          0          0  100      0      adic1:at:000012
no_recycle VSN             0          0          0   99      1      adic1:at:000026
no_recycle VSN             0          0          0   94      6      adic1:at:000025
no_recycle VSN             0          0          0   51     49      adic1:at:000020
no-data VSN                0          0          0   17     83      adic1:at:000017
no-data VSN                0          0          0   10     90      adic1:at:000018
empty VSN                  0          0          0    0    100      adic1:at:CLN003
.
.
.
```

3. 伺服器上若無使用中的資料歸檔到所選的 **VSN** 上，請使用 `tplabel(1M)` 或 `odlabel(1M)` 指令重新標記 **VSN**。

備註 – 這將損毀 **VSN** 上的所有資料，並重新恢復空間。

例如：對於磁帶 **VSN** 000029，請使用下列指令：

```
sky# tplabel -vsn 000029 -old 000029 at.000029
```

重新標記此 **VSN** 000029 後，您將重新取得該 **VSN** 上的所有空間。

如果媒體是磁光碟，請使用 `odlabel(1M)` 指令。如需有關 `odlabel(1M)` 指令的更多資訊，請參閱「`odlabel(1M)` 線上手冊」。

▼ 回收 partially full VSN

您也可以回收報告 partially full 狀態的 VSN。

1. 檢查用戶端中的 `recycler.log` 檔案，查看是否有任何 partially full VSN。

您可以考慮將用戶端 zeke 中的 VSN 000025 回收，因為其狀態為 partially full，如程式碼範例 7-25 中所示。

程式碼範例 7-25 用戶端 zeke 上的 `recycler.log` 檔案

```
# From the client zeke recycler.log file:
      ---Archives---      -----Percent-----      defaultset.2
-----Status-----      Count      Bytes      Use Obsolete Free      Library:Type:VSN
no-data VSN                0          0          0  100      0      skyrs:at:000029
no-data VSN                0          0          0   99      1      skyrs:at:000026
partially full            111        2.8G        6   88      6      skyrs:at:000025
empty VSN                  0          0          0    0     100      skyrs:at:000028
empty VSN                  0          0          0    0     100      skyrs:at:000027
```

VSN 000025 顯示了其 6% 的空間在使用中。必須重新歸檔這些使用中的歸檔影像，才能重新回收此 VSN。本程序中的下列步驟顯示了如何確定這些使用中的歸檔影像重新歸檔到其他 VSN。

2. 檢查伺服器端中的 `recycler.log` 檔案，以確定伺服器使用中的資料未歸檔在該 VSN 上。

伺服器的 `recycler.log` 檔案顯示了 VSN 000025 有 6% 的可用空間，這與用戶端的 `recycler.log` 檔案中報告的百分比相同。伺服器未偵測到用戶端的歸檔影像，因而報告剩餘的 94% 皆為過時的歸檔影像所佔用。

程式碼範例 7-26 伺服器 sky 上的 `recycler.log` 檔案

```
# From the Server log file:
      ---Archives---      -----Percent-----      adic1
-----Status-----      Count      Bytes      Use Obsolete Free      Library:Type:VSN
no-data VSN                0          0          0  100      0      adic1:at:000011
no_recycle VSN             0          0          0  100      0      adic1:at:000029
no-data VSN                0          0          0  100      0      adic1:at:000013
no-data VSN                0          0          0  100      0      adic1:at:000012
no_recycle VSN             0          0          0   99      1      adic1:at:000026
no_recycle VSN             0          0          0   94      6      adic1:at:000025
no_recycle VSN             0          0          0   51     49      adic1:at:000020
no-data VSN                0          0          0   17     83      adic1:at:000017
.
.
.
```

3. 在伺服器上使用 `chmed(1M)` 指令及 `+c` 選項重新將使用中的檔案歸檔到 VSN 上。

```
sky# chmed +c at.000025
```

如需有關 `chmed(1M)` 指令的更多資訊，請參閱「`chmed(1M)` 線上手冊」。

4. 在用戶端上使用 `sam-recycler(1M)` 指令再次執行回收器。

```
zeke# sam-recycler -dvx
```

這會將每一個使用中的檔案標記為要重新歸檔到其他 VSN 上。

5. 啟動歸檔器。

若要執行此作業，可以讓歸檔器正常執行，或在用戶端上的 `samu(1M)` 公用程式中鍵入 `:arrun`。如需有關 `:arrun(1M)` 指令的更多資訊，請參閱「`samu(1M)` 線上手冊」。

6. 歸檔完成後，請輸入 `sam-recycler(1M)` 指令，以在用戶端上重新執行回收器。

```
zeke# sam-recycler -dvx
```

這可確保所有使用中的檔案均已重新歸檔。

7. 若 `[Count]`、`[Bytes]` 與 `[Use]` 欄位值皆為 0 (零)，請使用 `tplabel(1M)` 或 `odlabel(1M)` 指令從伺服器重新標記 VSN。

以磁帶 VSN 000025 為例，請使用下列指令：

```
sky# tplabel -vsn 000025 -old 000025 at.000025
```

此指令會重新標記 VSN，並銷毀其中的所有資料。重新標記此 VSN 後，您將重新取得該 VSN 上 88% 的空間。

如果媒體是磁光碟，請使用 `odlabel(1M)` 指令。如需有關 `odlabel(1M)` 指令的更多資訊，請參閱「`odlabel(1M)` 線上手冊」。

於 Sun SAM-Remote 環境中進行回收 — 方法 2

本章節說明了您可以使用 Sun SAM-remote 軟體回收磁碟區的另一種方法。



注意 – 您必須完成本程序所有步驟並測試配置，以驗證回收是否正確執行之後，才可在 Sun SAM-Remote 環境中使用回收器。

▼ 配置回收 — 方法 2

1. 在 **Sun SAM-Remote** 用戶端上，輸入 `sam-recycler(1M)` 指令以判斷最佳回收候選者的磁碟區。

例如：

```
client# sam-recycler -dvx
```

2. 分析回收候選者的回收器記錄檔。

接近 `recycler.log` 檔案結尾的是 Status 欄。用戶端記錄檔中具有下列狀態項目類型的 VSN，即是要進行回收的候選者：

- `no-data` VSN。要回收 `no-data` VSN，請參閱第 184 頁的「回收 `no-data` VSN」。
- `partially full`。要回收 `partially full` VSN，請參閱第 186 頁的「回收 `partially full` VSN」。

在伺服器記錄檔中，最佳的回收候選者是 Count、Bytes 及 Use 欄中數值為 0 的項目。

3. 在 **Sun SAM-Remote** 伺服器上輸入 `chmed(1M)` 指令，以設定所選 VSN 的回收旗標。

例如：

```
server# chmed +c at.00025
```

4. 在 **Sun SAM-Remote** 用戶端上輸入 `sam-recycler(1M)` 指令，以回收 **Sun SAM-Remote** 用戶端上所選的 VSN。

例如：

```
client# sam-recycler -dvx
```

5. 請稍候，直到要回收的 VSN 完全清空歸檔影像。

用戶端上的歸檔器將執行此操作。

6. 在 **Sun SAM-Remote** 伺服器上輸入 `tplabel(1M)` 或 `odlabel(1M)` 指令（視歸檔媒體而定），以重新標記磁碟區。

7. 在 **Sun SAM-Remote** 伺服器上清除 `R` 或 `C` 等各式旗標，以防止將磁碟區用於在 **Sun SAM-Remote** 用戶端上進行歸檔。

進階主題

本章說明基本系統管理與用法範圍之外的進階主題。

本章包含以下各節。

- 第 189 頁的「在 Sun Cluster 環境中配置 SAM-QFS 檔案系統 (HA-SAM)」
- 第 195 頁的「使用裝置記錄」
- 第 197 頁的「使用可移除式媒體檔案」
- 第 199 頁的「使用區段檔案」
- 第 200 頁的「使用系統錯誤功能報告」

在 Sun Cluster 環境中配置 SAM-QFS 檔案系統 (HA-SAM)

高可用性 Sun StorageTek SAM (HA-SAM) 是 SAM-QFS 檔案系統和在基於 SPARC® 與 x64 硬體的 Solaris™ (SPARC 支援 Solaris 9 和 10, x64 僅支援 Solaris 10) 上執行的 Sun Cluster 軟體之間的介面。HA-SAM Sun Cluster 代理程式會定期監視 Sun StorageTek SAM 作業在主節點上的運作狀態。如果發生無法復原的問題，它會將 Sun StorageTek SAM 歸檔作業和分段備份作業切換至運作正常的節點。「主動-被動」配置支援自願和非自願的容錯移轉。僅支援兩個節點的「主動-被動」配置。

若要在容錯移轉之後繼續磁帶歸檔和分段備份，磁帶機必須對於 HA-SAM 執行所在之叢集中的所有節點均可見，但不應將其配置為 Sun StorageTek SAM 共用磁帶機。在磁碟歸檔對於叢集中的所有節點均可見的情況下，HA-SAM 也支援磁碟歸檔作業。

HA-SAM 依賴於 Sun StorageTek QFS Sun Cluster 代理程式，並假設 Sun StorageTek QFS 檔案系統由 Sun StorageTek QFS 代理程式掛載與管理。HA-SAM 需要將 Sun StorageTek SAM 目錄和分段備份程式目錄從標準位置連結至 HASToragePlus 檔案系統中的目錄。HA-SAM 應是包含 Sun StorageTek QFS 和目錄資源的資源群組中之資源。

必須已配置 Sun StorageTek QFS Sun Cluster 代理程式，才能配置 HA-SAM Sun Cluster 代理程式。這些指示假設您已按「Sun StorageTek QFS 安裝與升級指南」和「Sun StorageTek QFS 檔案系統配置與管理指南」中所述，在使用共用的 Sun StorageTek QFS 檔案系統之 Sun Cluster 環境中配置了 Sun StorageTek QFS。

備註 – HA-SAM 僅可搭配 Sun StorageTek SAM 和 Sun StorageTek QFS 軟體版本 4U6 和更新版本使用。

開始之前

配置此功能之前應注意下列需求和限制。

- 將在其之上配置 HA-SAM 的節點應執行相同版本的 Solaris (9 或 10) 和 Sun Cluster (3.1u4、3.2 或更新版本)。不支援 Solaris 或 Sun Cluster 軟體的混用版本。不支援 Solaris 以外的作業系統。如果是將版本 3.1u4 與 HA-SAM 搭配使用，請確定套用 Sun Cluster 軟體的最新修補程式。
- 執行 HA-SAM 的叢集中之節點必須有相同的架構類型：SPARC 或 x64。不支援混合架構。
- 不得配置兩個以上的節點。
- 僅支援「主動-被動」配置，不支援「主動-主動」配置。
- 僅在 HA-SAM 檔案系統的使用中節點上支援對 HA-SAM 檔案系統的主動 I/O。
- 僅支援共用的 Sun StorageTek QFS 檔案系統。支援 ma 型和 ms 型的檔案系統。不支援獨立式 Sun StorageTek QFS 環境。
- 此配置不支援軟體磁碟區管理員。
- 在 HA-SAM 環境內，必須為任何非 HA-SAM Sun StorageTek QFS 檔案系統指定 nosam 掛載選項。
- HA-SAM 資源、QFS 檔案系統和 HAStoragePlus 檔案系統均必須在同一資源群組內進行配置。必須為非 HA-SAM 檔案系統建立單獨的 Sun Cluster 資源群組。
- 使用 SUNW.hasam 資源類型時，不能在 /etc/vfstab 檔案中指定 bg 掛載選項。
- 需要光纖磁帶機。磁帶機必須透過光纖結構對所有系統均可見，但不應將其配置為 SAM-QFS 共用磁帶機。
- 用於進行磁碟歸檔的磁碟區必須對於所有節點均可見。
- 使用中的中介資料伺服器 and 潛在中介資料伺服器絕不能配置為 SAM-Remote 用戶端或伺服器。
- Sun StorageTek SAM 目錄和分段備份目錄必須位於預設位置：
/var/opt/SUNWsamfs/。如果目錄位於任何其他位置，叢集節點將找不到此目錄。
- 在配置 HA-SAM 之前，請驗證所有 Sun StorageTek SAM 作業在叢集中所需的節點上均運作正常。
- 僅支援高度可用的 (HA) 代理程式，不支援可延伸的代理程式。

- 此配置不支援 Oracle 軟體，但支援 HA-SAM 代理程式做為 Oracle 備份的 RMAN 目標。
- 自願容錯移轉會強制在 5 分鐘後執行容錯移轉，以避免其他潛在 HA 代理程式發生問題。
- 在自願容錯移轉期間，使用中的歸檔和分段備份會在 5 分鐘內逾時和終止。
- 在容錯移轉期間，所有 SAM-QFS 檔案系統均必須進行容錯移轉，以避免回收器故障而導致資料遺失。
- 支援 Sun StorageTek 網路連結式磁帶櫃，但不支援 ADIC 或 Sony 8400 PetaSite 系列自動磁帶櫃。
- File System Manager 瀏覽器介面無法管理 HA-SAM 環境。
- 在非自願容錯移轉期間，`/var/adm/messages` 記錄檔中可能會顯示閒置磁帶機的錯誤訊息。這些訊息無安全疑慮，可加以忽略。它們僅表示 HA-SAM 在容錯移轉之前沒有時間閒置磁帶機。
- 容錯移轉之後，目錄中的磁碟區會以 E 旗標標記。如果軟體無法於磁帶結尾處編寫正確的標籤，便會發生此情況。若要清除此錯誤，請參閱第 22 頁的「清除媒體錯誤」。

配置指示

配置 HA-SAM 之前，您必須先使用 Sun Cluster `scrgadm` 指令配置下列延伸特性：

- `QFSName` — 提供逗號分隔的共用 QFS 系列集名稱清單
- `CatalogFileSystem` — 指定在 `/etc/vfstab` 中指定的叢集 HAStoragePlus 檔案系統

備註 — 如需有關使用 `scrgadm` 指令的詳細資訊，請參閱 Sun Cluster 管理指南。

▼ 驗證 SUNWhasam 資源類型註冊檔案

- 使用下列指令可驗證 SUNWhasam 資源類型註冊 (RTR) 檔案是否正確連結：

```
# ls -l /usr/cluster/lib/rgm/rtreg/SUNW.hasam
lrwxrwxrwx 1 root  other  32 Feb 26 10:59
/usr/cluster/lib/rgm/rtreg/SUNW.hasam ->
/opt/SUNWsamfs/sc/etc/SUNW.hasam
```

▼ 確認目錄和分段備份程式符號連結

SAM-QFS 目錄和分段備份程式佇列必須位於 HAStoragePlus 檔案系統中，才能配置為高可用性。在軟體安裝程序期間，應已建立從 /var/opt/SUNWsamfs 到共用 HAStoragePlus 檔案系統的符號連結。您必須確認已連結，才能繼續配置程序。

- 驗證 /var/opt/SUNWsamfs/catalog 和 /var/opt/SUNWsamfs/stager 是否包含至 HAStoragePlus 掛載點的符號連結。

- a. 檢視目錄和分段備份檔案。

```
# ls -l /var/opt/SUNWsamfs/catalog /var/opt/SUNWsamfs/stager
```

- b. 驗證輸出是否顯示類似以下範例的符號連結。

```
lrwxrwxrwx 1 root  other    19 Sep 30 11:05
/var/opt/SUNWsamfs/catalog -> /sam_shared/catalog
lrwxrwxrwx 1 root  other    18 Sep 30 11:05
/var/opt/SUNWsamfs/stager -> /sam_shared/stager
```

如果存在符號連結，請進入第 193 頁的「在 Sun Cluster 環境中配置 Sun StorageTek SAM」。如果連結不存在，請進入步驟 c。

- c. 將所有現有的 Sun StorageTek SAM 目錄資訊複製到暫存位置。
- d. 從目前位置刪除現有的 catalog 和 stager 檔案。

```
# rm -rf /var/opt/SUNWsamfs/catalog /var/opt/SUNWsamfs/stager
```

- e. 建立至 HAStoragePlus 掛載點的 /var/opt/SUNWsamfs/catalog 和 /var/opt/SUNWsamfs/stager 符號連結。

以下範例顯示掛載於 /sam_shared 的 HAStoragePlus 檔案系統。

```
# ln -s /sam_shared/catalog /var/opt/SUNWsamfs/catalog
# ln -s /sam_shared/stager /var/opt/SUNWsamfs/stager
```

- f. 將步驟 c 中儲存的目錄資訊複製到新共用目錄。

```
# cp -rp /var/tmp/catalog/* /var/opt/SUNWsamfs/catalog
```


▼ 在 Sun Cluster 環境中配置 Sun StorageTek SAM

在此範例程序中，會配置兩個檔案系統：qfs1 和 qfs2。

1. 建立 HA-SAM 資源群組。

```
# scrgadm -a -g SAM-HA -h scnode-A,scnode-B
```

2. 建立並配置 SUNW.qfs 資源。

```
# scrgadm -a -g SAM-HA -t SUNW.qfs -j qfs-res \  
-x QFSFileSystem=/global/qfs1,/global/qfs2
```

3. 建立並配置 Sun StorageTek SAM 共用資源。

```
# scrgadm -a -j sam-hastp -g SAM-HA -t SUNW.HAStoragePlus \  
-x FilesystemMountPoints=/sam_shared -x AffinityOn=TRUE
```

4. 建立 HA-SAM 資源類型。

```
# scrgadm -a -t SUNW.hasam -f /opt/SUNWsamfs/sc/etc/SUNW.hasam
```

5. 建立並配置 SUNW.hasam 資源。

```
# scrgadm -a -g SAM-HA -t SUNW.hasam -j sam-ha \  
-x QFSName=qfs1,qfs2 -x CatalogFileSystem=/sam_shared
```

6. 建立資源群組內資源之間的相依性。

```
# scrgadm -c -j qfs-res -y Resource_dependencies=sam-hastp  
# scrgadm -c -j sam-ha -y Resource_dependencies=qfs-res
```

7. 使用 `scswitch(1M) -Z -g` 指令將資源群組變成線上狀態。

```
# scswitch -Z -g SAM-HA
```

備註 – 必須先掛載 Sun StorageTek QFS 檔案系統，然後才能將資源群組變成線上狀態。

8. 確定資源群組在所有已配置的節點上可運作。

```
# scswitch -z -g SAM-HA -h scnode-B  
# scswitch -z -g SAM-HA -h scnode-A
```

使用 `samd hastop` 指令

`samd` 指令有一個專門用於 HA-SAM 的選項。`samd hastop` 指令用於在停止 `sam-aml` 及其子項之前，先停止歸檔器和分段備份程式常駐程式。

在獨立式或共用 Sun StorageTek SAM 環境中，管理員均不得使用 `samd hastop` 指令。此外，一般的 `samd stop` 指令也不得在 HA-SAM 所控制的節點上使用。

搭配 HA-SAM 使用 Sun StorageTek 程式庫

如果在 HA-SAM 環境中使用 Sun StorageTek 程式庫，需要在發生非自願容錯移轉時，通知 STK 常駐程式使用強制卸載而非一般卸載。若要完成此作業，需要在將 HA-SAM 資源變成線上狀態時，於使用中的 Sun Cluster 節點上建立 `/var/run/hasam_running` 檔案。此為具有 `root` 權限的零位元檔案。如果在某個節點上偵測到此檔案，`sam-stkd` 和 `sam-stk_helper` 會使用 `force_media` 卸載磁帶機上的磁帶。這會在非自願容錯移轉之後，成功地初始化程式庫和磁帶機。將節點上的 SAM 資源變成離線狀態時，會刪除 `hasam_running` 檔案。

為確保 Sun StorageTek 程式庫的容錯移轉運作方式正確，請在發現 Sun Cluster 節點上有 HA-SAM 資源處於線上狀態時，確定未刪除 `/var/run/hasam_running`。

使用裝置記錄

裝置記錄功能提供可用於分析某些類型裝置問題的指定裝置之錯誤資訊。它可協助決定自動程式庫、磁帶機或光碟機的事件失敗順序。裝置記錄功能不會收集軟式媒體錯誤(如可回復的讀取錯誤)。

裝置記錄訊息將寫入個別記錄檔。每個自動程式庫、磁帶與光碟機裝置及記錄器均有一個記錄檔。記錄檔位於 `/var/opt/SUNWsamfs/devlog`。每一個記錄檔的名稱皆會對應於設備序號的名稱。

例如，假設您擁有一個 Sun StorageTek SAM 檔案系統與一個含有兩部光碟機的 Hewlett Packard 光碟程式庫。

程式碼範例 8-1 顯示 mcf 檔案。

程式碼範例 8-1 mcf 檔案範例

```
/dev/samst/c1t5u0 40 hp hp40 - etc/opt/SUNWsamfs/hp40_cat
/dev/samst/c1t4u0 41 mo hp40 -
/dev/samst/c1t6u0 42 mo hp40 -
```

程式碼範例 8-2 顯示 `/var/opt/SUNWsamfs/devlog` 檔案。

程式碼範例 8-2 devlog 檔案

```
# pwd
/var/opt/SUNWsamfs/devlog
# ls
40      41      42      43
#
```

裝置 43 是記錄器。

何時使用裝置記錄檔

裝置記錄檔可輕易產生許多記錄檔訊息，特別是在所有裝置的全部記錄選項已開啓並且裝置活動頻繁時。最初的裝置記錄設定已設爲以下預設值：

```
err retry syserr date
```

如果懷疑 Sun StorageTek SAM 環境中配置的某個裝置有問題，啓動該裝置的額外記錄檔事件不失爲適當的方法。此外，如果服務供應商建議您啓用裝置記錄檔，您也可以這麼做。在這類情況下，將事件設爲 `detail`。在極端情況下，服務供應商可能會建議您將裝置的事件設爲 `all`。這將新增額外的記錄檔資訊。但一般來說，使用過多的記錄檔執行系統可能不太有用或切實可行。

裝置記錄檔資訊會在輸入 `samexplorer(1M)` 指令時自動收集。這可讓檔案系統服務將任何可能的裝置錯誤資訊做爲問題分析活動的一部分來檢閱。

啓用裝置記錄檔

您可以透過下列各節所述的兩種方式之一啓用裝置記錄檔：

- [第 196 頁的「使用 `samset\(1M\)` 指令以啓動裝置記錄檔」](#)
- [第 196 頁的「編輯 `defaults.conf` 檔案以啓動裝置記錄檔」](#)

▼ 使用 `samset(1M)` 指令以啓動裝置記錄檔

- 使用 `samset(1M)` 指令

例如：

```
# samset devlog eq event
```

對於 `eq`，請指定要記錄訊息之裝置的設備序號。

對於 `event`，請指定「`samset(1M)` 線上手冊」中所列的一或多個事件。如果您指定了一個以上的事件，請以空格字元區隔事件。

▼ 編輯 `defaults.conf` 檔案以啓動裝置記錄檔

1. 成爲超級使用者。
2. 使用 `vi(1)` 或其他編輯器開啟 `/etc/opt/SUNWsamfs/defaults.conf` 檔案。

3. 將 `devlog` 指令增加到 `defaults.conf` 檔案中。

```
devlog eq event
```

對於 `eq`，請指定要記錄訊息之裝置的設備序號。

對於 `event`，請指定「`samset(1M)` 線上手冊」中所列的一或多個事件。如果您指定了一個以上的事件，請以空格字元區隔事件。

當 Sun StorageTek SAM 檔案系統啟動時，它會自動將每個可用裝置的事件類型設為 `default`。您也可以使用 `samset(1M)` 指令來決定每個裝置記錄檔的目前設定。

4. 儲存並關閉 `defaults.conf` 檔案。
5. 使用 `samd(1M) config` 指令來散佈 `defaults.conf` 檔案變更。

```
# samd config
```

使用可移除式媒體檔案

您可使用 `request(1)` 指令手動建立、寫入及讀取不使用磁碟快取記憶體來緩衝資料的檔案。以此方式建立的檔案稱為可移除式媒體檔案。

備註 – `request(1)` 指令會略過歸檔器的一般功能。

可移除式媒體檔案因為具有權限、使用者名稱、群組名稱及大小屬性而顯得與一般的 Sun StorageTek SAM 檔案相似。但其資料並不存在於磁碟快取記憶體中。因此，您可以建立大於磁碟快取記憶體的可移除式媒體檔案，並將其寫入可移除式媒體卡匣中。

系統會針對您在 `request(1)` 指令上指定的檔案，在 `.inodes` 檔案中建立 `inode` 項目。Sun StorageTek SAM 檔案系統會從 `inode` 項目讀取資訊。多個可移除式媒體檔案可位於相同磁碟區中。

如果可移除式媒體檔案橫跨多個磁碟區，即此檔案稱為磁碟區溢位檔案。`volume overflow` 功能可讓單一大型檔案橫跨於多個卡匣的多個磁碟區上。如果您有超出選定媒體容量之極大檔案，磁碟區溢位功能則會很有用。

您必須依序讀取和寫入可移除式媒體檔案。如果磁碟區位於一個 `mcf` 檔案中所定義的自動程式庫，Sun StorageTek SAM 檔案系統會自動掛載請求的磁碟區。

磁碟區如果存在可移除式媒體檔案，則可防止回收該磁碟區。回收器僅預期歸檔檔案位於指定用於歸檔的特定磁碟區。此外，可移除式媒體檔案將永不歸檔。NFS 不支援可移除式媒體檔案。

▼ 建立可移除式媒體或磁碟區溢位檔案

1. 使用 `tplabel(1M)` 或 `odlabel(1M)` 指令分別標記磁帶或磁光碟匣。
要取得有關這些指令的資訊，請參閱其個別線上手冊。
2. 使用 `request(1)` 指令
至少，使用以下選項：

```
request -m media-type -v vsn [vsn/vsn ...] [-l vsn-file] input-file
```

表 8-1 `request(1)` 指令的引數

引數	意義
<i>media-type</i>	可移除式媒體卡匣的媒體類型。如需有關有效 <i>media-type</i> 指定的資訊，請參閱「 <code>mcf(4)</code> 線上手冊」。
<i>vsn</i>	可移除式媒體卡匣的磁碟區序列名稱 (VSN)。 您若是指定多個 VSN，便會建立磁碟區溢位檔案。您最多可以為磁碟區溢位檔案指定 256 個 VSN。使用斜線字元 (/) 來區隔 <i>vsn</i> 引數。 指定的 VSN 不應位於 Sun StorageTek SAM 環境中用以進行自動歸檔的磁碟區之間。歸檔時，會將下一個要歸檔的檔案附加至目前資料的尾端，並將 EOF 標籤移到資料之後。
<i>vsn-file</i>	含有 VSN 清單的輸入檔案。您如有多個 VSN，則在輸入檔案中指定 VSN 清單，會比在指令行中進行指定容易。
<i>input-file</i>	要寫入可移除式媒體卡匣的檔案。此檔案必須位於 Sun StorageTek SAM 檔案系統中。

以下指令會建立可移除式媒體檔案：

```
# request -m lt -v aaa rem1
```

以下指令會在三個磁碟區上建立磁碟區溢位檔案：

```
# request -m lt -v TAPE01/TAPE02/TAPE03 large.file
```

如需有關如何建立可移除式媒體檔案的詳細範例，請參閱「`request(1)` 線上手冊」。

使用區段檔案

Sun StorageTek SAM 環境支援分段的檔案。區段檔案可提高超大容量檔案的磁帶儲存裝置之取得速度、存取及易管理性。區段檔案的容量可超過實體磁碟快取記憶體。不論何時，區段檔案皆可能只保留一部分在磁碟快取記憶體中。

`segment(1)` 指令可讓您指定區段的大小。無法將區段大小設為小於目前檔案的大小。

區段檔案支援磁帶資料平行儲存區。在檔案分段之後，可將它同時串列於多個磁帶裝置，這樣可大量減少儲存檔案區段所需的時間。允許使用者僅取得所需的檔案區段而非整個檔案，可加速資料的存取。

由於僅歸檔檔案中變更的部分，因此分段可提高歸檔效率。檔案中的區段可同時歸檔，區段檔案可同時分段備份。如此將可提升歸檔與擷取期間的效能。

分段可在檔案、目錄或整個檔案系統中啟動。區段檔案可支援所有其他的 Sun StorageTek SAM 功能。

備註 – `mmap` 功能無法在區段檔案上執行。因此，區段檔案無法用為可執行的二進位檔案。

以下章節說明區段檔案與非區段檔案之間的區別。如需有關區段檔案的更多資訊，請參閱「`segment(1)` 或 `sam_segment(3)` 線上手冊」。

歸檔

對於區段檔案，可歸檔單位指的是區段本身而非檔案。所有歸檔屬性與優先權將套用於個別區段而非檔案。

您可以在 `archiver.cmd` 檔案中為歸檔組指定 `-drives` 和 `-drivemin` 參數，以平行儲存區段。例如，假設檔案系統中有一個 100 MB 的區段檔案，其區段大小為 10 MB。如果 `archiver.cmd` 檔案使用 `-drives 2` 指令定義一個歸檔集，則此檔案將平行歸檔至兩個磁碟機。區段 1、3、5、7 及 9 使用第一台磁碟機歸檔，區段 2、4、6、8 及 10 則使用第二台磁碟機歸檔。

只有經過修改的區段會進行歸檔。每個區段最多可建立四個歸檔備份。Sun StorageTek SAM 也可支援區段的磁碟區溢位。

備註 – 區段檔案的索引不包含使用者資料。其會被視為中介資料，並會指定到檔案系統歸檔組。

災難復原

若需關於發生災難時復原區段檔案的資訊，請參閱「Sun StorageTek Storage Archive Manager 疑難排解指南」。

使用系統錯誤功能報告

系統錯誤功能 (SEF) 報告系統會擷取自動程式庫中磁帶裝置的記錄感應資料，將其寫入記錄檔，然後轉譯成可閱讀的形式。其由下列內容組成：

- 一個包含磁帶裝置記錄檔感應頁資料的記錄檔。
- 一個 `sefreport(1M)` 指令，用於將記錄檔以人類可讀取的格式寫入 `stdout`。此記錄檔可用作輸入使用者提供的分析指令碼。

各廠商提供的記錄檔感應頁會不同。有關參數代碼、控制位元及參數值的含義，請參閱每個指定裝置的供應商文件資料。

獨立式磁帶機不支援 SEF。對於不支援 `tapealert(1M)` 功能的舊型 SCSI-2 裝置，SEF 報告是最有用的。要取得更多相關資訊，請參閱「`tapealert(1M)` 線上手冊」。

▼ 啟動 SEF 報告

1. 成為超級使用者。
2. 使用 `mkdir(1)` 指令建立 SEF 目錄。

例如：

```
# mkdir /var/opt/SUNWsamfs/sef
```

3. 使用 `touch(1)` 指令建立記錄檔。

透過建立 `sefdata` 記錄檔，可在安裝之後隨時啟動 SEF 報告。最初，SEF 記錄檔必須是空的。

下列指令會顯示要在預設位置中建立的 SEF 記錄檔。

```
# touch /var/opt/SUNWsamfs/sef/sefdata
```

4. 使用 `samd(1M) stop` 和 `samd(1M) start` 初始化 SEF 報告。

```
# samd stop  
# samd start
```

SEF 資料將在產生之後附加至記錄檔。

備註 – 如有 `sefdata` 記錄檔存在，即會啟用 SEF 報告。若要停用 SEF 報告，必須重新命名此檔案，或予以移除。

您可將 SEF 報告設定為從替代位置記錄與讀取記錄檔感應資料。如需有關從替代位置讀取記錄檔感應資料的更多資訊，請參閱「`sefreport(1M)` 線上手冊」。

▼ 產生 SEF 報告輸出

在使用 `sefreport(1M)` 指令之前，確定 `/opt/SUNWsamfs/sbin` 位於指令路徑中。SEF 報告輸出由標題行與記錄檔感應資料組成。

- 使用 `sefreport(1M)` 指令來產生 SEF 輸出。

以下選項是 `sefreport(1M)` 指令中最常用的選項：

- `-d` 選項：`-d` 選項產生額外的裝置資訊。其可將每筆記錄中含有設備序號與路徑名稱的額外標頭行寫入裝置。這樣可輕易搜尋並找出有關指定裝置的 SEF 記錄。

- `-v` 選項或 `-t` 選項：

- `-v` 選項以詳細資訊模式產生資訊。此選項可將有關設備序號、頁面代碼及 VSN 的資訊附加至記錄的每一行。這樣則可能僅選擇有關指定裝置或指定欄的文字行。

- `-t` 選項會產生含有文字說明的記錄檔感應輸出。報告會針對記錄檔感應資料輸出的各行併入額外的字串，於其中提供設備序號、頁面代碼、VSN 及參數碼說明等資訊。

- 請勿在相同的指令行上指定 `-t` 和 `-v` 選項。它們會彼此排除。

例如，下列 SEF 指令可從預設位置讀取 SEF 記錄檔，寫入每個裝置的裝置號碼與路徑名稱，然後產生執行結果：

```
# sefreport -d /var/opt/SUNWsamfs/sef/sefdata > sef.output
```

程式碼範例 8-3 顯示了 `sef.output` 檔案的內容。

程式碼範例 8-3 `sef.output` 內容

```
Record no. 1
Mon Mar 26 11:17:48 2001  STK          9840          1.25 VSN 002981
  Eq no. 32   Dev name /dev/rmt/1cbn
  PAGE CODE 2
  param code  control   param value
    00h       74h       0x0
    01h       74h       0x0
    02h       74h       0x0
    03h       74h       0x0
    04h       74h       0x0
    05h       74h       0x40050
    06h       74h       0x0
```

程式碼範例 8-3 sef.output 內容 (續)

```
PAGE CODE 3
param code  control  param value
    00h      74h     0x0
    01h      74h     0x0
    02h      74h     0x0
    03h      74h     0x0
    04h      74h     0x0
    05h      74h     0x140
    06h      74h     0x0
PAGE CODE 6
param code  control  param value
    00h      74h     0x0

Record no. 2
Mon Mar 26 11:30:06 2001  STK          9840          1.25 VSN 002999
Eq no. 31   Dev name /dev/rmt/0cbn
PAGE CODE 2
param code  control  param value
    00h      74h     0x0
    01h      74h     0x0
    02h      74h     0x0
    03h      74h     0x0
    04h      74h     0x0
    05h      74h     0x1400a0
    06h      74h     0x0

PAGE CODE 3
param code  control  param value
    00h      74h     0x0
    01h      74h     0x0
    02h      74h     0x0

    03h      74h     0x0
    04h      74h     0x0
    05h      74h     0x190
    06h      74h     0x0
PAGE CODE 6
param code  control  param value
    00h      74h     0x0
```

```

Record no. 3
Mon Mar 26 11:30:23 2001 STK          9840          1.25 VSN 002981
  Eq no. 32   Dev name /dev/rmt/1cbn
  PAGE CODE 2
    param code  control  param value
      00h       74h     0x0
      01h       74h     0x0
      02h       74h     0x0
      03h       74h     0x0
      04h       74h     0x0
      05h       74h     0x18400f0
      06h       74h     0x0

  PAGE CODE 3
    param code  control  param value
      00h       74h     0x0
      01h       74h     0x0
      02h       74h     0x0
      03h       74h     0x0
      04h       74h     0x0
      05h       74h     0x1e0
      06h       74h     0x0

  PAGE CODE 6
    param code  control  param value
      00h       74h     0x0
  .
  .
  .

```

如需有關 SEF 記錄檔的更多資訊 (包含其內容與格式)，請參閱「sefdata(4) 線上手冊」。如需有關選擇使用的 SEF 報告格式之更多資訊，請參閱「sefreport(1M) 線上手冊」。

管理 SEF 記錄檔

管理 SEF 記錄檔的方式和管理任何其他 Sun StorageTek SAM 記錄檔一樣。您可以定期執行 cron(1) 工作，將目前的記錄檔儲存至其他位置、刪除舊的 SEF 檔案、建立新的 (空的) SEF 檔案或執行其他檔案管理作業。

您也可以使用 log_rotate.sh(1M) 公用程式以循環此記錄檔。

如需有關管理 SEF 記錄檔的工具之更多資訊，請參閱「cron(1) 線上手冊」或「log_rotate.sh(1M) 線上手冊」。

SEF sysevent 功能

除可使用 SEF 記錄檔之外，還可使用 Solaris sysevent 功能取得磁帶機 SCSI 記錄檔感應錯誤計數器的第 2 頁與第 3 頁，以進行媒體分析。SEF sysevent 功能預設為啓用，並會設定成在卸載前執行一次輪詢。SEF sysevent 運作方式是由 defaults.conf 和 samset 控制。

您可以在 defaults.conf 檔案中使用 sef 參數，依設備序號啓用 SEF sysevent 功能，或指定記錄檔感應輪詢頻率。如需更多資訊，請參閱「defaults.conf(4) 線上手冊」。

▼ 建立 SEF sysevent 處理程式

1. 使用類似於下列指令建立 /var/tmp/xx 檔案：

```
#!/bin/ksh
echo "$@" >> /var/tmp/xx.dat
exit 0
```

2. 使 /var/tmp/xx 檔案可執行：

```
# chmod a+rxw /var/tmp/xx
```

3. 透過建入下列指令，將 SEF sysevent 處理程式增加至 syseventd 檔案：

```
# syseventadm add -vSUNW -pSUNWsamfs -cDevice -sSEF
/var/tmp/xx "\$VENDOR\" "\$PRODUCT\" "\$USN\" "\$REV\" \
$TOD \$EQ_ORD "\$NAME\" \$INQ_TYPE "\$MEDIA_TYPE\" "\$VSN\"
\$LABEL_TIME \$LP2_PC0 \$LP2_PC1 \$LP2_PC2 \$LP2_PC3 \$LP2_PC4
\$LP2_PC5 \$LP2_PC6 \$LP3_PC0 \$LP3_PC1 \$LP3_PC2 \$LP3_PC3 \
\$LP3_PC4 \$LP3_PC5 \$LP3_PC6 \$WHERE \$sequence
# syseventadm restart
```

此指令可建立內含 SEF sysevent 處理程式 /var/tmp/xx 的 /etc/sysevent/config/SUNW,SUNWsamfs,Device,sysevent.conf 檔案，並可將事件處理程式載入 syseventd 常駐程式中。

備註 – Solaris 8 作業系統不支援 syseventadm(1M) 指令。您必須手動建立 /etc/sysevent/config/SUNW,SUNWsamfs,Device,sysevent.conf 檔案與內容，然後鍵入 `kill -HUP syseventd`，才可重新啓動 syseventd 常駐程式。

4. 若要載入 SEF sysevent 處理程式，請使用 `kill -HUP syseventd` 指令啟動 /var/tmp/xx SEF sysevent 處理程式。

如需有關 SEF sysevent 用法的更多資訊，請參閱「sefsysevent(4) 線上手冊」。

使用供應商指定作業程序的程式庫 基本作業

您可以將許多不同製造商的程式庫包含在 Sun StorageTek SAM 環境中。針對大部分的程式庫，您應該使用第 11 頁的「使用自動程式庫和手動載入磁碟機」中說明的作業程序。但某些程式庫具有廠商指定的作業程序，這些作業程序將在本章中說明。

備註 – Sun StorageTek SAM 軟體可以與許多製造商的自動程式庫相容。請諮詢 Sun 銷售代表或是授權的服務供應商，以獲得有關程式庫型號、韌體等級，以及其他相容性資訊。

本章將說明以下自動程式庫：

- 第 206 頁的「ADIC/Grau 自動程式庫」
- 第 207 頁的「Fujitsu LMF 自動程式庫」
- 第 209 頁的「IBM 3584 UltraScalable 磁帶櫃」
- 第 210 頁的「IBM 3494 程式庫」
- 第 211 頁的「Sony 8400 PetaSite 直接連結式自動程式庫」
- 第 214 頁的「Sony 網路連結式自動程式庫」
- 第 215 頁的「StorageTek ACSLS 連結式自動程式庫」

ADIC/Grau 自動程式庫

如果擁有 ADIC/Grau 自動程式庫，請使用本節的程序以匯入與匯出卡匣。這些程序與第 11 頁的「使用自動程式庫和手動載入磁碟機」中說明的不同。

因為您實際上使用供應商提供的公用程式，來新增和移除 ADIC/Grau 自動程式庫中的卡匣，Sun StorageTek SAM 介面 (`import(1M)`、`samexport(1M)` 和 File System Manager) 只會影響程式庫目錄檔。

備註 – 在 x64 硬體平台上，Sun StorageTek SAM 軟體不支援 ADIC/Grau 網路連結式程式庫。

▼ 匯入卡匣

要匯入卡匣，請執行以下步驟：

1. 使用 **ADIC/Grau** 指令以將卡匣實際移至程式庫。
2. 使用 **Sun StorageTek SAM** `import(1M)` 指令更新程式庫目錄。

請用以下格式使用此指令：

```
import -v volser eq
```

表 A-1 `import(1M)` 指令的引數

引數	意義
<i>volser</i>	欲新增的 <i>volser</i> 。在使用新項目更新程式庫目錄檔之前， <code>grauaci</code> 介面會確認 ADIC/Grau 自動程式庫具有 <i>volser</i> 資訊。
<i>eq</i>	如 <code>mcf</code> 檔案中所定義要定址之裝置的設備序號。

▼ 匯出卡匣

要匯出卡匣，請執行以下步驟：

1. 使用 **Sun StorageTek SAM** `samexport(1M)` 指令從程式庫目錄檔移除項目。
以下列格式之一使用此指令：

```
samexport eq:slot  
samexport media-type.vsn
```

表 A-2 samexport(1M) 指令的引數

引數	意義
<i>eq</i>	如 <code>mcf</code> 檔案中所定義要定址之裝置的設備序號。
<i>slot</i>	表示在程式庫目錄檔中識別的自動程式庫中儲存插槽的號碼。
<i>media-type</i>	表示媒體類型。如需有效媒體類型清單，請參閱「 <code>mcf(4)</code> 線上手冊」。
<i>vsn</i>	指定給磁碟區的磁碟區序列名稱 (VSN)。

`samexport(1M)` 指令會在匯入每個 VSN 時更新程式庫目錄，並將每個 VSN 的程式庫目錄項目從程式庫目錄移至記錄器。

2. 使用 **ADIC/Grau** 指令以將卡匣實際移出程式庫。

Fujitsu LMF 自動程式庫

如果擁有 Fujitsu LMF 自動程式庫，請使用本節的程序以匯入與匯出卡匣。這些程序與第 11 頁的「[使用自動程式庫和手動載入磁碟機](#)」中說明的不同。

因為您實際上使用供應商提供的公用程式，來新增或移除 Fujitsu LMF 自動程式庫中的卡匣，Sun StorageTek SAM 介面 (`import(1M)`、`samexport(1M)` 和 File System Manager) 只會影響程式庫目錄檔。

備註 – 在 x64 硬體平台上，Sun StorageTek SAM 軟體不支援 Fujitsu LMF 網路連結式程式庫。

▼ 匯入卡匣

要匯入卡匣，請執行以下步驟：

1. 使用 **Fujitsu** 指令以將卡匣實際移至程式庫。
2. 使用 **Sun StorageTek SAM import(1M)** 指令更新程式庫目錄。

請用以下格式使用此指令：

```
import -v volser eq
```

表 A-3 import(1M) 指令的引數

引數	意義
<i>volser</i>	欲新增的 <i>volser</i> 。在使用新項目更新程式庫目錄檔之前， <i>fujitsulmf</i> 介面會確認 LMF 自動程式庫具有 <i>volser</i> 資訊。
<i>eq</i>	如 <i>mcf</i> 檔案中所定義要定址之裝置的設備序號。

▼ 匯出卡匣

要匯出卡匣，請執行以下步驟：

1. 使用 **Sun StorageTek SAM samexport(1M)** 指令從程式庫目錄檔移除項目。
以下列格式之一使用此指令：

```
samexport eq:slot  
samexport media-type.vsn
```

表 A-4 samexport(1M) 指令的引數

引數	意義
<i>eq</i>	如 <i>mcf</i> 檔案中所定義要定址之裝置的設備序號。
<i>slot</i>	表示在程式庫目錄檔中識別的自動程式庫中儲存插槽的號碼。
<i>media-type</i>	表示媒體類型。如需有效媒體類型清單，請參閱「 <i>mcf(4)</i> 線上手冊」。
<i>vsn</i>	指定給磁碟區的磁碟區序列名稱 (VSN)。

samexport(1M) 指令會在匯入每個 VSN 時更新程式庫目錄檔，並將每個 VSN 的程式庫目錄項目從 Sun StorageTek SAM 程式庫目錄檔移至 Sun StorageTek SAM 記錄器。

2. 使用 **Fujitsu** 指令以將卡匣實際移出程式庫。

IBM 3584 UltraScalable 磁帶櫃

Sun StorageTek SAM 環境中支援 IBM 3584 UltraScalable 磁帶櫃。以下章節說明此程式庫作業與第 11 頁的「使用自動程式庫和手動載入磁碟機」說明程序不同的地方。

備註 – 在 x64 硬體平台上，Sun StorageTek SAM 軟體不支援 IBM 3584 UltraScalable 程式庫。

匯入卡匣

當啟動 Sun StorageTek SAM 軟體時，並不會自動匯入信箱中的卡匣。

清潔磁碟機

若要在 Sun StorageTek SAM 環境中使用此程式庫，請停用自動清潔然後啟動主控清潔。IBM 刊物 GA32-0408-01「IBM 3584 UltraScalable 磁帶櫃規劃與操作指南」中有說明本程序。「ibm3584(7) 線上手冊」中也有說明。

分割

此程式庫可容納幾個磁碟機。如果使用多個磁碟機，則可能會將此一個實體程式庫切分成兩個、三個或四個邏輯程式庫。如果已將程式庫切分成兩個或更多的邏輯程式庫，在將 IBM 3584 程式庫新增到 Sun StorageTek SAM 環境之前，請確定這些邏輯程式庫操作正常。

當卡匣從已分割的程式庫中匯出時，僅有匯出該卡匣的邏輯程式庫可存取拔取器插槽。如果移除並手動重新插入卡匣，則任何邏輯分割區均可存取卡匣。

▼ 移除卡匣

若要從分割程式庫移除卡匣，請執行下列步驟。

1. 開啟門蓋。
2. 移除卡匣。
3. 關閉門蓋。

4. 等待門蓋鎖定，然後解除鎖定。
5. 開啟門蓋。
6. 更換卡匣。
7. 關閉門蓋。

如需使用本程式庫當做 Sun StorageTek SAM 環境中之邏輯分割程式庫的詳細資訊，請參閱 IBM 文件資料或「ibm3584(7) 線上手冊」。

IBM 3494 程式庫

Sun StorageTek SAM 環境中支援 IBM 3494 程式庫。以下章節說明此程式庫作業與第 11 頁的「使用自動程式庫和手動載入磁碟機」說明程序不同的地方。

備註 – 在 x64 硬體平台上，Sun StorageTek SAM 軟體不支援 IBM 3494 網路連結式程式庫。

▼ 匯入卡匣

要匯入卡匣，請執行以下步驟：

1. 將新的媒體置入 I/O 插槽。
2. 關閉門蓋。

程式庫會鎖定門蓋然後將媒體移至儲存區內。您一次只能匯入 100 個磁碟區。

如果程式庫配置為 `access=private`，則這是您需要執行的最後一步。程式庫會通知常駐程式媒體已經移動，並且新增至目錄檔中。

3. 若將程式庫配置為 `access=shared`，請輸入 `import(1M)` 指令，將媒體增加到目錄中。

▼ 匯出卡匣

1. 使用 `export(1M)` 指令匯出卡匣。

本指令將媒體移至 I/O 區域，並開啓作業面板上的輸出模式燈號。

2. 實際將媒體從 I/O 區域移除。

Sony 8400 PetaSite 直接連結式自動程式庫

Sony 8400 PetaSite 系列自動程式庫與其他 Sony 型號不同，因為它有一個八個插槽的匯入與匯出信箱 (插槽 400–407)。由於此原因，此系統中的匯入與匯出作業更為簡單直接。此自動程式庫使用條碼讀取器。

因為信箱插槽可以用來當做儲存插槽，Sun StorageTek SAM 程式庫目錄檔會持續追蹤信箱插槽。

備註 – 在 x64 硬體平台上，Sun StorageTek SAM 軟體不支援 Sony 8400 PetaSite 程式庫。

備註 – 本節中的資訊僅適用於 Sony 8400 PetaSite 直接連結式自動程式庫。本資訊與 Sony 直接連結式 B9 和 B35 自動程式庫無關，同時也與第 214 頁的「[Sony 網路連結式自動程式庫](#)」無關。

▼ 匯入磁帶

要匯入磁帶，請按照以下步驟執行。

1. 按下自動程式庫前面板上的開啟/關閉按鈕，以開啟自動程式庫的門蓋。
2. 將卡匣載入信箱插槽。
3. 按下自動程式庫前面板上的開啟/關閉按鈕，然後手動關閉信箱的門蓋。

在關閉門蓋之後，自動程式庫會檢查信箱插槽的卡匣條碼。如果條碼有問題，該插槽的 in 和 out 指示燈都將會閃爍。

4. 使用 `import(1M)` 指令使 Sun StorageTek SAM 系統能夠識別匯入的卡匣。

請用以下格式使用此指令：

```
import eq
```

對於 `eq`，請依據 `mcf` 檔案所定義，指定要定址之裝置的設備序號。

您也可以使用 File System Manager 執行此步驟。若需更多資訊，請參閱 File System Manager 線上說明。

匯出磁帶

匯出磁帶卡匣的程序將視是否將信箱插槽用作儲存插槽而定。

▼ 未將信箱插槽用為儲存插槽時匯出磁帶

當沒有將信箱插槽用作儲存插槽時，使用以下程序匯出卡匣。

1. 輸入 **move(1M)** 指令以將卡匣移至電子信箱插槽 (插槽 400-407)。

請用以下格式使用此指令：

```
move source-slot destination-slot eq
```

表 A-5 move(1M) 指令的引數

引數	意義
<i>source-slot</i>	表示卡匣目前所連接插槽的插槽號碼。
<i>destination-slot</i>	表示不應將卡匣移至插槽的插槽號碼。
<i>eq</i>	如 <i>mcf</i> 檔案中所定義要定址之裝置的設備序號。

2. 按下自動程式庫前面板上的開啟/關閉按鈕。
門蓋將會開啓。
3. 移除信箱插槽中的卡匣。
4. 按下自動程式庫前面板上的開啟/關閉按鈕，然後手動關閉信箱的門蓋。
5. 輸入 **samexport(1M)** 指令使 **Sun StorageTek SAM** 系統能夠識別匯出的卡匣。
請用以下格式使用此指令：

```
samexport eq
```

對於 *eq*，請依據 *mcf* 檔案所定義，指定要定址之裝置的設備序號。

您也可以使用 **File System Manager** 執行此步驟。如需更多資訊，請參閱 **File System Manager** 線上說明。

▼ 將信箱插槽用為儲存插槽時匯出磁帶

在將信箱插槽用作儲存插槽，並且要匯出的卡匣在其中一個信箱插槽中時，使用以下程序匯出卡匣。

1. 按下自動程式庫前面板上的開啟/關閉按鈕。
門蓋將會開啓。
2. 移除信箱插槽中的卡匣。
3. 按下自動程式庫前面板上的開啟/關閉按鈕，然後手動關閉信箱的門蓋。
4. 輸入 `samexport(1M)` 指令使 **Sun StorageTek SAM** 系統能夠識別匯出的卡匣。
請用以下格式使用此指令：

```
samexport eq
```

對於 `eq`，請依據 `mcf` 檔案所定義，指定要定址之裝置的設備序號。

您也可以使用 **File System Manager** 執行此步驟。如需更多資訊，請參閱 **File System Manager** 線上說明。

▼ 將卡匣移至不同插槽

若要將卡匣移至不同插槽，請執行下列步驟。

1. 確定來源插槽已被佔用，並且目的插槽是空的。
2. 輸入 `move(1M)` 指令。
請用以下格式使用此指令：

```
move eq:source-slot destination-slot
```

表 A-6 `move(1M)` 指令的引數

引數	意義
<code>eq</code>	如 <code>mcf</code> 檔案中所定義要定址之裝置的設備序號。
<code>source-slot</code>	表示卡匣目前所連接插槽的插槽號碼。
<code>destination-slot</code>	表示不應將卡匣移至插槽的插槽號碼。

您也可以使用 **File System Manager** 執行此步驟。如需更多資訊，請參閱 **File System Manager** 線上說明。

Sony 網路連結式自動程式庫

若是使用 Sony 網路連結式自動程式庫，請使用本節的程序匯入與匯出卡匣。這些程序與第 11 頁的「使用自動程式庫和手動載入磁碟機」中說明的不同。

因為您實際上使用供應商提供的公用程式，來新增和移除 Sony 自動程式庫中的卡匣，Sun StorageTek SAM 介面 (`import(1M)`、`samexport(1M)` 和 `File System Manager`) 只會影響程式庫目錄。

備註 – 在 x64 硬體平台上，Sun StorageTek SAM 軟體不支援 Sony 網路連結式程式庫。

▼ 匯入卡匣

要匯入卡匣，請執行以下步驟：

1. 使用 `Sony` 指令以將卡匣實際移至程式庫。
2. 使用 `import(1M)` 指令更新程式庫目錄。

請用以下格式使用此指令：

```
import -v [ " ] volser [ " ] eq
```

表 A-7 `import(1M)` 指令的引數

引數	意義
" "	表示引號。 <code>volser</code> 值中若含有空格，必須以引號括起。
<code>volser</code>	欲新增的 <code>volser</code> 。在使用新項目更新程式庫目錄之前，PSC API 介面會驗證 Sony 自動程式庫是否包含 <code>volser</code> 資訊。如果卡匣並非在程式庫中實際存在，項目將儲存在記錄器目錄中。
<code>eq</code>	如 <code>mcf</code> 檔案所定義要定址之程式庫的設備序號。

▼ 匯出卡匣

要匯出卡匣，請執行以下步驟：

1. 使用 `samexport(1M)` 指令從程式庫目錄移除項目。

以下列格式之一使用此指令：

```
samexport eq:slot  
samexport media-type.vsn
```

表 A-8 samexport(1M) 指令的引數

引數	意義
<i>eq</i>	如 <code>mcf</code> 檔案中所定義要定址之裝置的設備序號。
<i>slot</i>	表示在程式庫目錄檔中識別的自動程式庫中儲存插槽的號碼。
<i>media-type</i>	表示媒體類型。如需有效媒體類型清單，請參閱「 <code>mcf(4)</code> 線上手冊」。
<i>vsn</i>	指定給磁碟區的磁碟區序列名稱 (VSN)。

`samexport(1M)` 指令會在匯入每個 VSN 時更新程式庫目錄，並將每個 VSN 的程式庫目錄項目從程式庫目錄移至記錄器。

2. 使用 `Sony` 指令以將卡匣實際移出程式庫。

StorageTek ACSLS 連結式自動程式庫

如果擁有 StorageTek ACSLS 網路連結式自動程式庫，請使用本節的程序以匯入與匯出卡匣。這些程序與第 11 頁的「使用自動程式庫和手動載入磁碟機」中說明的不同。

信箱是自動程式庫中用以放入及移除卡匣的區域。某些 StorageTek 自動程式庫一次只能匯入與匯出一個卡匣。StorageTek 9714 和 StorageTek 9710 皆是 Sun StorageTek SAM 環境所支援含有信箱的 StorageTek 自動程式庫。StorageTek 9730 則是使用郵件插槽。在 StorageTek 文件中，信箱與信箱插槽通常稱為卡匣存取連接埠 (CAP)。

當從 ACSLS 連結式自動程式庫匯入和匯出卡匣時，請注意以下事項：

- 匯入卡匣時，Sun StorageTek SAM 指令只會對程式庫目錄造成影響。`import(1M)` 指令不會將卡匣實際插入自動程式庫。必須使用 ACSLS 指令才能實際匯入卡匣。
- 匯出卡匣時，除非同時在 `samexport(1M)` 指令中使用了 `-f` 選項，否則 Sun StorageTek SAM 指令只會對程式庫目錄造成影響。若是使用 `-f` 選項，將會指示 Sun StorageTek SAM 系統將磁碟區置於 CAP 中，並據此更新目錄。如果不指定 `-f` 選項，目錄會更新，但是因為磁碟區並未置於 CAP 中，所以仍然必須使用 ACSLS 指令實際匯出卡匣。

您的責任是持續將 ACSLS 庫存與 Sun StorageTek SAM 目錄保持一致。

您也可以使用 `samu(1M)` 或 `File System Manager` 執行匯入與匯出程序。如需更多資訊，請參閱 `File System Manager` 線上說明。

▼ 匯入磁帶

- 請以下列格式使用 `import(1M)` 指令：

```
import -v vsn eq
```

表 A-9 `import(1M)` 指令的引數

引數	意義
<i>vsn</i>	指定給磁碟區的磁碟區序列名稱 (VSN)。
<i>eq</i>	如 <code>mcf</code> 檔案中所定義要定址之裝置的設備序號。

`import(1M)` 指令會導致新的 VSN 出現在程式庫目錄中。如果記錄器中已存在 VSN，Sun StorageTek SAM 軟體會將 VSN 資訊從記錄器移至程式庫目錄。

▼ 使用信箱匯出磁帶

可使用插槽或 VSN 匯出磁帶卡匣。

- 請以下列一種格式使用 `samexport(1M)` 指令：

```
samexport [-f] eq:slot  
samexport [-f] media-type .vsn
```

表 A-10 `samexport(1M)` 指令的引數

引數	意義
<code>-f</code>	指定 Sun StorageTek SAM 系統將磁碟區置於卡匣存取連接埠 (CAP)，並據以更新目錄。
<i>eq</i>	如 <code>mcf</code> 檔案中所定義要定址之裝置的設備序號。
<i>slot</i>	表示在程式庫目錄檔中識別的自動程式庫中儲存插槽的號碼。
<i>media-type</i>	表示媒體類型。如需有效媒體類型清單，請參閱「 <code>mcf(4)</code> 線上手冊」。
<i>vsn</i>	指定給磁碟區的磁碟區序列名稱 (VSN)。

`samexport(1M)` 指令會在匯入每個 VSN 時更新程式庫目錄，並將每個 VSN 的程式庫目錄項目從程式庫目錄移至記錄器。

字彙表

英文字母

- DAU** 磁碟分配單元。線上儲存裝置的基本單元。也稱為區段大小。
- FDDI** 光纖分散式資料介面 (FDDI) 是區域網路中的資料傳輸標準，其範圍最多可延伸至 200 公里 (124 英里)。FDDI 通訊協定以記號環網路通訊協定為基礎。
- FTP** 檔案傳輸協定。在兩個主機之間透過 TCP/IP 網路傳送檔案的網際網路通訊協定。
- inode 檔案** 一個位於檔案系統上的特別檔案 (.inodes)，包含該檔案系統上所有檔案的 inode 結構。Inode 的長度是 512 位元組。Inode 檔案是中介資料檔案，與檔案系統中的檔案資料分隔開。
- inode** 索引節點。檔案系統用來描述檔案的資料結構。inode 描述與檔案關聯的所有屬性，不包含名稱。屬性包含所有權、存取、權限、大小及檔案在磁碟系統上的位置。
- LAN** 區域網路。
- LUN** 邏輯單元號碼。
- mc f** 主要配置檔案。這是在初始化時間讀取的檔案，此檔案將定義一個檔案系統環境中的裝置之間的關係 (拓撲)。
- NFS** 網路檔案系統。由 Sun 發行的一種檔案系統，提供連接到異質網路上的遠端檔案系統之透明存取。
- NIS** Sun OS 4.0 (至少) Network Information Service (網路資訊服務)。一種分散式網路資料庫，包含有關網路上的系統和使用者的主要資訊。NIS 資料儲存在主要伺服器 and 所有從屬伺服器上。
- RAID** 備援磁碟陣列。一種磁碟技術，使用數個獨立磁碟以穩定地儲存檔案。它可以保護資料免於單一磁碟故障所造成的資料遺失，它可以提供容錯磁碟環境，而且可以提供比個別磁碟更高的流量。

RPC	遠端程序呼叫。NFS 所使用的基本資料交換機制，以實作自訂的網路資料伺服器。
SAM-QFS	將 Sun StorageTek SAM 軟體與 Sun StorageTek QFS 檔案系統結合的配置。SAM-QFS 為使用者與管理員提供高速、標準的 UNIX 檔案系統介面，以搭配儲存與歸檔管理公用程式使用。它使用 Sun StorageTek SAM 指令集中的許多指令，以及標準 UNIX 檔案系統指令。
samfsdump	一個程式，可建立控制結構傾印，並為特定檔案群組複製所有的控制結構資訊。它與 UNIX tar(1) 公用程式類似，但是它通常不會複製檔案資料。另請參閱 <i>samfsrestore</i> 。
samfsrestore	一個程式，可從控制結構傾印復原 inode 與目錄資訊。另請參閱 <i>samfsdump</i> 。
SCSI	小型電腦系統介面。一種電子通訊規格，一般用於週邊裝置，如磁碟機、磁帶機及自動程式庫。
Sun SAM-Remote	
伺服器	既是全容量 Sun StorageTek SAM 儲存管理伺服器，也是定義要在 Sun SAM-Remote 用戶端之間共用程式庫的 Sun SAM-Remote 伺服器常駐程式。
Sun SAM-Remote	
用戶端	具有用戶端常駐程式的 Sun StorageTek SAM 系統，包含一些虛擬裝置，也可以擁有本身的程式庫裝置。用戶端需要 Sun SAM-Remote 伺服器才能為一個或多個歸檔備份進行媒體歸檔。
tar	磁帶歸檔。歸檔影像使用的標準檔案和資料記錄格式。
TCP/IP	傳輸控制協定/網際網路協定。網際網路協定負責主機對主機的定址與路由、封包傳送 (IP) 及在應用程式點 (TCP) 之間穩定傳送資料。
vfstab 檔案	vfstab 檔案包含檔案系統的掛載選項。指令行上指定的掛載選項會覆寫 /etc/vfstab 檔案中指定的掛載選項，但 /etc/vfstab 檔案中指定的掛載選項則會覆寫 <i>samfs.cmd</i> 檔案中指定的掛載選項。
VSN	磁碟區序列名稱。若要歸檔到可攜式媒體卡匣，VSN 是指寫入到磁碟區標籤的磁帶與光碟之間的邏輯識別碼。若要歸檔到磁碟快取記憶體，這是磁碟歸檔組的唯一名稱。
WORM	寫入一次，多次讀取。一種媒體儲存裝置歸類，指只能寫入一次，但可以讀取多次的媒體。

一畫

乙太網路 一種本機區域、封包交換的網路技術。原先的設計使用同軸電纜，現在都使用遮蔽式雙絞線。乙太網路是每秒 10 或 100 MB 的 LAN。

三畫

小型電腦系統介面 請參閱 SCSI。

四畫

中介資料 有關資料的資料。中介資料是用來尋找磁碟上的檔案之正確資料位置的索引資訊。它包含有關檔案、目錄、存取控制清單、符號連結、可攜式媒體、區段檔案及區段檔案索引等資訊。

中介資料裝置 一個儲存檔案系統中介資料的裝置 (例如：實體狀態磁碟或鏡像裝置)。將檔案資料和中介資料儲存在不同的裝置上可以增進效能。在 `mcf` 檔案中，中介資料裝置會宣告為 `ma` 檔案系統中的 `mm` 裝置。

分段備份 從歸檔儲存裝置將近線或離線檔案複製回線上儲存裝置的程序。

分割區 裝置的一部分，或磁光碟匣的一面。

五畫

主機檔案 主機檔案包含共用檔案系統中所有主機的清單。若要将檔案系統初始化為 Sun StorageTek QFS 共用檔案系統，必須先建立主機檔案 `/etc/opt/SUNWsamfs/hosts.fs-name`，再建立此檔案系統。`sammkfs(1M)` 指令在建立該檔案系統時會使用此主機檔案。您可以在稍後使用 `samsharefs(1M)` 指令替代或更新主機檔案的內容。

卡匣 包含用於錄製資料的媒體之實際實體，例如磁帶或光碟。有時亦稱為媒體、磁碟區或媒介。

可定址儲存裝置 包含使用者透過 Sun StorageTek QFS 或 Sun StorageTek SAM 檔案系統來參照的線上、近線、遠端及離線儲存裝置的儲存空間。

可移除的媒體檔案	一種特殊使用者檔案類型，可以直接在其常駐的可攜式媒體卡匣上存取，如磁帶或光碟匣。也用來寫入歸檔與分段備份檔案資料。
本機檔案系統	安裝在 Sun Cluster 系統的某個節點上的檔案系統，而且不對其他節點提供高度可用性。此外，也是一個安裝在伺服器上的檔案系統。
用戶端伺服器	一個分散式系統互動模式，在此模式下，一方的程式傳送請求給另一方的程式，然後等待回應。發出請求的程式稱為用戶端。滿足回應的程式稱為伺服器。
目錄	指向檔案系統內其他檔案與目錄的檔案資料結構。
目錄檔	自動程式庫內的 VSN 記錄。每個自動程式庫都有一個目錄檔，而在一個站點中，所有自動程式庫會有一個記錄器。另請參閱 VSN。

六畫

光纖通道	ANSI 標準，指定裝置之間的高速序列通訊。光纖通道是 SCSI-3 內的一個匯流排架構。
全域指令	適用於所有檔案系統，且出現在第一個 fs = 行之前的歸檔器與釋放器指令。
共用主機檔案	當您建立共用檔案系統時，系統會將主機檔案的資訊複製到中介資料伺服器上的共用主機檔案。您可以輸入 <code>samsharefs(1M) -u</code> 指令以更新此資訊。
名稱空間	一組檔案的中介資料部分，可以識別出檔案、檔案屬性及儲存位置。
回收器	一種 Sun StorageTek SAM 公用程式，可重新取得卡匣上由過期的歸檔備份所佔用的空間。
多重讀取器檔案系統	這是一種單一寫入器、多重讀取器的功能，可讓您指定一個可掛載在多個主機上的檔案系統。多部主機可以讀取該檔案系統，但是只有一部主機可以寫入檔案系統。要指定多重讀取器，請使用 <code>mount(1M)</code> 指令及其 <code>-o reader</code> 選項。要指定單一寫入器主機，則使用 <code>mount(1M)</code> 指令及其 <code>-o writer</code> 選項。如需有關 <code>mount(1M)</code> 指令的更多資訊，請參閱「 <code>mount_samfs(1M)</code> 線上手冊」。
自動程式庫	一種自動裝置，可自動載入與卸載可攜式媒體卡匣，不需操作者介入。一個自動程式庫包含一個或多個磁帶機，而且含有傳輸機制，可以在儲存插槽與磁帶機之間來回移動卡匣。
自動裝置	在儲存插槽與磁帶機之間移動卡匣的自動程式庫組成部分。又稱為「transport」（傳輸器）。

七畫

- 系列組** 一個儲存裝置，代表一個獨立實體裝置群組，如一組磁碟或自動程式庫內的磁帶機。另請參閱**儲存裝置系列組**。
- 系統裝置組** 請參閱系列組。

八畫

- 延伸陣列** 在一個檔案的 **inode** 之內的陣列，定義每個指定給檔案的資料區段之磁碟位置。
- 直接 I/O** 用於大型區段對齊循序式 I/O 的屬性。**setfa(1)** 指令的 **-D** 選項為直接 I/O 選項。它會將檔案或目錄設成直接 I/O 屬性。如果套用於某個目錄，直接 I/O 屬性將會被繼承。
- 直接存取** 檔案屬性 (永不分段備份)，表示近線檔案可以直接從歸檔媒體中存取，不需要擷取到磁碟快取記憶體。
- 直接連結式程式庫** 使用 **SCSI** 介面直接與伺服器連接的自動程式庫。**SCSI** 連結式程式庫直接由 **Sun StorageTek SAM** 軟體控制。
- 近線儲存裝置** 必須先自動掛載才可以存取的可攜式媒體儲存裝置。近線儲存裝置通常比線上儲存裝置的價格便宜，不過它將耗用更多的存取時間。

九畫

- 計時器** 配額軟體，記錄使用者達到軟式限制的開始時間，以及對使用者實行強制限制的結束時間。

十畫

- 核心** 提供基本系統功能的中央控制程式。**UNIX** 核心會建立與管理程序，提供存取檔案系統的功能、提供一般性安全性，以及供應通訊功能。
- 租約** 一個讓用戶端主機取得在特定的時間內對某個檔案執行作業的權限的功能。中介資料伺服器會發出租約給每個用戶端主機。若要繼續進行檔案作業，您必須更新租約。
- 配額** 使用者獲准使用的系統資源量。

十一畫

區塊大小 請參閱 *DAU*。

區段分配對映 一個點陣圖，呈現磁碟上的每個可用儲存區段，並指出區段已被使用或尚未被佔用。

強制限制 若是磁碟配額，這是使用者不能超過的檔案系統資源、區段和 *inode* 的最大限制。

掛載點 掛載檔案系統的目錄。

軟式限制 在磁碟配額上，使用者可以暫時超過的檔案系統資源 (區塊與 *inodes*) 的臨界值限制。超過軟式限制的話會啟動計時器。當您超過軟式限制並到達所指定的時間，除非您減少檔案系統的使用率到軟式限制以下，否則將無法分配更多的系統資源。

連線 兩個協定模組之間的路徑，提供穩定的串流傳送服務。*TCP* 連線會從一部機器的 *TCP* 模組延伸到另一部的 *TCP* 模組。

十二畫

備份儲存 一組檔案的快照，以避免資料突然遺失。備份包含檔案的屬性與相關資料。

媒體 磁帶或光碟卡匣。

媒體回收 對具有很少使用中檔案的歸檔媒體進行回收或再使用的程序。

循環式 資料存取方式，這種方式會將整個檔案以循序的方式寫入邏輯磁碟上。當單一檔案寫入磁碟時，整個檔案會寫入第一個邏輯磁碟。第二個檔案會寫入下一個邏輯磁碟，依此類推。每個檔案的大小決定 *I/O* 的大小。

請同時參閱磁碟資料平行儲存以及資料平行儲存。

程式庫 請參閱自動程式庫。

程式庫目錄檔 請參閱目錄檔。

虛擬裝置 沒有相關硬體的軟體子系統或驅動程式。

超級區塊 檔案系統中定義檔案系統基本參數的資料結構。超級區塊將寫入到儲存裝置系列組的所有分割區，並識別該組中的分割區成員身份。

間接區段 包含一系列儲存區段的磁碟區段。檔案系統擁有最多三個層級的間接區段。第一層間接區段包含一系列供資料儲存使用的區段。第二層間接區段包含一系列第一層間接區段。第三層間接區段包含一系列第二層間接區段。

十三畫

- 裝置記錄** 一種可配置的功能，提供裝置專用的錯誤資訊，以便分析裝置問題。
- 裝置掃描器** 這是會定期監視所有手動掛載的可攜式裝置是否存在，並可由使用者或其他程序要求以偵測已掛載卡匣是否存在的軟體。
- 資料平行儲存** 資料存取方式，這種方式會以交錯的方式，同步將檔案寫入邏輯磁碟。**SAM-QFS** 檔案系統提供兩種資料平行儲存方式：「強制資料平行儲存」，使用資料平行儲存群組；以及「軟式資料平行儲存」，使用 `stripe=x` 掛載參數。設定檔案系統並且要求在 `mcf` 檔案內定義資料平行儲存群組時，即會啟用強制資料平行儲存。軟式資料平行儲存則是透過 `stripe=x` 掛載參數啟用，而且可以就檔案系統或個別檔案做變更。設定 `stripe=0` 即可停用。若是檔案系統由多個具有相同元素數量的資料平行儲存群組組成，則強制與軟式資料平行儲存皆可使用。另請參閱循環式。
- 資料平行儲存區大小** 在寫入到下一個資料平行儲存裝置以前，所要分配的磁碟分配單元 (DAU) 數量。若是使用 `stripe=0` 掛載選項，檔案系統使用循環存取，而非資料平行儲存存取。
- 資料平行儲存群組** 檔案系統內的裝置集合，其在 `mcf` 檔案中定義為一個或多個 `gXXX` 裝置。資料平行儲存群組可視為一個邏輯裝置，通常會將資料平行儲存成與磁碟分配單元 (DAU) 的大小一樣。
- 資料裝置** 在檔案系統中負責儲存檔案資料的一個裝置或一組裝置。
- 預先分配** 在磁碟快取記憶體上預留連續空間的程序，供寫入檔案使用。僅有其大小為零的檔案可以指定預先分配。如需更多資訊，請參閱「`setfa(1)` 線上手冊」。

十四畫

- 磁帶機** 一個可以在可攜式媒體磁碟區中來回傳送資料的機制。
- 磁碟快取記憶體** 檔案系統軟體中的磁碟常駐部分，用來建立和管理線上磁碟快取記憶體與歸檔媒體之間的資料檔案。個別的磁碟分割區或一整個磁碟都可以用來做為磁碟快取記憶體。
- 磁碟空間臨界值** 磁碟快取記憶體使用率的最大或最小層級，如管理者所定義者。釋放器會依據這些預先定義的磁碟空間臨界值，控制磁碟快取記憶體使用率。
- 磁碟分配單元** 請參閱 *DAU*。
- 磁碟區** 卡匣上供共用資料使用的命名區域。一個卡匣有一個或多個磁碟區。雙面卡匣有兩個磁碟區，一面一個。

磁碟區溢位	讓系統將單一檔案跨越到多個磁碟區的功能。在站點使用超過其個別卡匣容量的大型檔案時，磁碟區溢位是很有幫助的。
磁碟資料平行儲存	將檔案記錄到數個磁碟的程序，因此增進存取效能並提高整體的儲存容量。另請參閱資料平行儲存。
磁碟緩衝區	在使用 Sun SAM-Remote 軟體時，磁碟緩衝區是伺服器系統上的一塊緩衝區，供資料從用戶端歸檔至伺服器時使用。
網路連結式自動程式庫	使用供應商所提供的套裝軟體加以控制的程式庫，如那些來自 StorageTek、ADIC/Grau、IBM 或 Sony 者。Sun StorageTek SAM 檔案系統可以使用特別為自動程式庫設計的 Sun StorageTek SAM 媒體抽換裝置常駐程式做為與供應商軟體之間的介面。
遠端程序呼叫	請參閱 RPC。
遠端儲存裝置	距離伺服器較遙遠的儲存裝置，用於災難復原。

十五畫

寬限期	若是磁碟配額，這是允許使用者在到達軟式限制以後建立檔案和分配儲存空間的時間量。
稽核 (完整)	載入卡匣以驗證其 VSN 的程序。對磁光碟匣來說，會判斷其容量與空間資訊，並放進自動程式庫目錄檔中。另請參閱 VSN。
線上儲存裝置	立即可使用的儲存裝置，例如磁碟快取記憶體裝置。

十七畫

儲存插槽	磁帶機卡匣未在磁碟機中使用時在自動程式庫內的位置。
儲存裝置系列組	一組磁碟，共同以單一磁碟系列裝置的形態呈現。
檔案系統	檔案與目錄的階層式集合。
檔案系統特定指令	跟隨 archiver.cmd 檔案中的全域指令之歸檔器和釋放器指令，為特定檔案系統的專屬指令，且以 fs = 開頭。在遇到下一個 fs = 指令行、或檔案結束之前，檔案系統專用指令均有效。如果有數個影響檔案系統的指令，則檔案系統特定指令會覆寫全域指令。

十八畫

- 歸檔媒體** 歸檔檔案所寫入的媒體。歸檔媒體可以是可攜式磁帶或程式庫中的磁光碟匣。此外，歸檔媒體可以是另外一個系統上的掛載點。
- 歸檔器** 自動控制將檔案複製到可攜式卡匣的歸檔程式。
- 歸檔儲存** 建立在歸檔媒體上的檔案資料副本。
- 離線儲存裝置** 需要操作者介入才能載入的儲存裝置。

十九畫

- 鏡像寫入** 在無關的磁碟組上為檔案保存兩份副本的程序，以防止因其中一個磁碟發生故障而導致資料遺失。

二十畫

- 釋放器** Sun StorageTek SAM 元件，指出被歸檔的檔案並釋放其磁碟快取記憶體副本，因此可以得到更多可用的磁碟快取記憶體空間。釋放器會根據臨界值的高低自動對線上磁碟儲存裝置總量進行管制。
- 釋放優先權** 在檔案系統中已歸檔的檔案被釋放的優先權。釋放優先權是根據檔案特性的不同權數之相乘，然後將結果加總來計算。

索引

英文字母

- access 歸檔器指令, 57
- ACSAPI 介面, 3
- ACSLs 連結式程式庫, 215
- ADIC/Grau 自動程式庫
 - 作業, 206
- age_priority preview.cmd 指令, 127
- allsets 歸檔組, 33, 67
- archivemeta 歸檔器指令, 46
- archiver(1M) 指令, 42, 164
 - 範例, 91
 - 輸出範例, 39
- archiver.cmd 檔案, 32, 90
 - access 和 -nftv 指令, 57
 - archivemeta 指令, 46
 - archmax 指令, 47, 67
 - bufsize 指令, 48, 68
 - drivemax, drivemin, 和磁碟機歸檔請求指令, 68
 - drives 指令, 49
 - endparams 指令, 66
 - fillvsns 歸檔請求參數, 70
 - fs 指令, 55
 - ignore 指令, 142, 144
 - join 路徑參數, 72
 - logfile 指令, 51
 - minsize 和 maxsize 指令, 58
 - norelease 指令, 64
 - offline_copy 參數, 71
 - ovflmin 指令, 52
 - params 指令, 66
 - recycle_dataquantity 指令, 143
 - recycle_ignore 指令, 142, 143, 144
 - recycle_mailaddr 指令, 143
 - recycle_mingain 指令, 143
 - SAM-Remote 範例, 170
 - startage, startcount, 及 startsize 參數, 78
 - tapenonstop 參數, 73
 - vsns 關聯指令, 79
 - vsnpools 參數, 81
 - vsns 和 endvsns 參數, 80
 - 分段備份中的角色, 125
 - 全域指令, 46
 - 名稱指令, 59
 - 回收指令, 71, 134
 - 多重中介資料副本, 66
 - 自動取消歸檔, 65
 - 使用 SAM-Remote 配置回收, 168
 - 使用者和群組指令, 59
 - 保留參數, 74
 - 建立, 42
 - 指令, 43, 45
 - 配置回收器以進行磁碟歸檔, 143
 - 通知指令, 51
 - 等候指令, 54
 - 間隔指令, 50
 - 資料平行儲存區段檔案, 199
 - 範例, 44
 - 編輯 SAM-Remote, 161
 - 複製數目指令, 63
 - 優先權參數, 77

- 檔案系統指令, 55
- 檢查指令, 49
- 歸檔時間, 65
- 歸檔組指定, 56
- 簡介, 42
- 鎖定歸檔請求參數, 70
- 釋放和分段備份指令, 61
- 釋放的角色, 115
- 釋放指令, 64
- archiver.sh 程序檔, 51
- archmax 歸檔器指令, 47, 67
- auditslot(1M) 指令, 17, 22
- bufsize 分段備份程式指令, 121
- bufsize 歸檔器指令, 48, 68
- chmed(1M) 指令, 20, 22, 139, 187, 188
- cleandrive(1M) 指令, 21
- crontab 項目, 回收器, 141, 144
- defaults.conf 檔案
 - exported_media 指令, 26, 29
 - 有人照料指令, 26, 29
 - 啓用裝置記錄檔, 196
- devlog 指令, 197
- devlog 檔案, 195
- diskvols.conf 檔案, 83
- display_all_candidates 釋放器指令, 112
- drivemax 歸檔器指令, 68
- drivemin 歸檔器指令, 68
- drives 分段備份程式指令, 120
- drives 歸檔組參數指令, 68
- drives 歸檔器指令, 49
- DZC-8000S 介面, 3
- endparams 歸檔器指令, 66
- endvsnpools 歸檔器指令, 81
- examine 歸檔器指令, 49
- export(1M) 指令, 210
- exported_media 指令, 26
- File System Manager
 - 建立帳號, 6
 - 搭配 sam-nrecycler 使用, 145
 - 管理遠端伺服器, 9
 - 簡介, 5
- fillvsns 歸檔請求參數, 70
- fs 歸檔器指令, 55
- fs 釋放器指令, 112
- Fujitsu LMF 自動程式庫作業, 207
- group 歸檔器指令, 59
- HA-SAM
 - 介紹, 4
 - 配置, 189
- HASStoragePlus 檔案系統, 190
- hlwm_priority_preview.cmd 檔案指令, 128
- hwm 回收器指令, 139
- hwm_archive 掛載選項, 50
- hwm_priority_preview.cmd 檔案指令, 128
- IBM 3494 自動程式庫
 - 作業, 210
- IBM 3584 自動程式庫
 - 分割, 209
 - 作業, 209
 - 清理, 209
- idle 指令, 13
- ignore 回收器指令, 139, 142, 144
- import(1M) 指令, 18, 24, 26, 206, 208, 210, 211, 214, 216
- interval 歸檔器指令, 50
- join path 歸檔器指令, 72
- lhwm_priority_preview.cmd 檔案指令, 128
- list_size 釋放器指令, 115
- lmcpd 介面, 3
- load_notify.sh 程序檔, 28
- lock 歸檔器指令, 70
- log_rotate.sh 程序檔, 203
- Logfile 分段備份程式指令, 122
- Logfile 歸檔器指令, 51
- Logfile 釋放器指令, 113
- lwm_priority_preview.cmd 檔案指令, 128
- mail 回收器指令, 139
- maxactive 分段備份程式指令, 125
- maxsize 歸檔器指令, 58
- mcf 檔案, 3
 - SAM-Remote 配置, 150
 - 程式庫記錄器, 25

- min_residence_age 釋放器指令, 113
- mingain 回收器指令, 139
- minsize 歸檔器指令, 58
- mount(1M) 指令
 - 部分釋放和分段備份選項, 106
- move(1M) 指令, 212, 213
- name 歸檔器指令, 59
- NFS 檔案共用, 148
- nftv 歸檔器指令, 57
- no_archive 歸檔組, 33, 57
- no_recycle 回收器指令, 136, 171
- no_release 釋放器指令, 112
- no-data VSN, 184
- norelease 歸檔器指令, 64
- notify 歸檔器指令, 51
- nrecycler.cmd 檔案, 145
- o maxpartial 掛載選項, 106
- o partial_stage 掛載選項, 106
- o 部分掛載選項, 106
- odlabel(1M) 指令, 16, 185, 187, 188, 198
- offline_copy 歸檔器指令, 71
- ovflmin 歸檔器指令, 52
- params 歸檔器指令, 66
- partially full VSN, 186
- pkginfo(1M) 指令, 153
- pool 歸檔器指令, 79
- preview.cmd 檔, 126
 - age_priority 指令, 127
 - hlwm_priority 指令, 128
 - hwm_priority 指令, 128
 - lhwm_priority 指令, 128
 - lwm_priority 指令, 128
 - vsn_priority 指令, 127
 - 指令, 126
 - 設定優先, 129
- priority 歸檔器指令, 77
- queue_time_limit 參數, 39
- research_no_release 釋放器指令, 114
- recycle_dataquantity 歸檔器指令, 143
- recycle_hwm 歸檔器指令, 169
- recycle_ignore 歸檔器指令, 142, 143, 144, 169
- recycle_mailaddr 歸檔器指令, 143
- recycle_mingain 歸檔器指令, 143, 169
- recycle_minopbs 百分比回收器指令, 85
- recycle_vsncount 歸檔器指令, 169
- recycler.cmd 檔案
 - hwm 指令, 139
 - ignore 指令, 139, 142, 144
 - logfile 指令, 135
 - mail 指令, 139
 - mingain 指令, 139
 - no_recycle 指令, 136
 - SAM-Remote, 165
 - 建立, 138
 - 配置 SAM-Remote, 170, 171
 - 程式庫指令, 136
 - 範例, 138
- recycler.sh 程序檔, 142, 172
- release 歸檔器指令, 61, 64
- release(1) 指令, 103
 - 部分釋放, 108
- releaser.cmd 檔案, 109, 116
 - display_all_candidates 指令, 112
 - fs 指令, 112
 - list_size 指令, 115
 - logfile 指令, 113
 - min_residence_age 指令, 113
 - no_release 指令, 112
 - research_no_release 指令, 114
 - weight_age 指令, 110
 - weight_age_access 指令, 110
 - weight_age_modify 指令, 110
 - weight_age_residence 指令, 110
 - weight_size 指令, 111
 - 檔案時間指令, 109
- request(1) 指令, 134, 197
 - 引數, 198
- reserve 歸檔器指令, 74
- sam_release(3) 程式庫常式, 108
- sam-amld 常駐程式, 126
- sam-archiverd 常駐程式, 32, 36 - 38
- sam-arcopy 程序, 32
 - 記錄檔, 40

- sam-arfind 程序, 32, 34
 - 記錄檔, 40
- samcmd(1M) 指令, 13
 - 卸載選項, 15, 27 - 28
 - 開啓選項, 13
 - 閒置選項, 30, 155
 - 載入選項, 14
 - 稽核選項, 17
 - 關閉選項, 14
- samd hastop 指令, 使用 HA-SAM, 194
- samd(1M) 指令
 - 停止選項, 155, 194
 - 啓動選項, 12, 161
- samexport(1M) 指令, 24, 27, 207, 208, 212, 215, 216
- samfsdump(1M) 指令, 90
- sam-genericd 常駐程式, 3
- sam-ibm3494d 常駐程式, 3
- sam-nrecycler(1M) 指令, 145
- sam-recycler(1M) 指令, 134, 135, 140, 144, 172, 175, 187, 188
- SAM-Remote
 - samu(1M) R 畫面, 163
 - 用戶端和伺服器互動, 151
 - 用戶端配置檔案, 157
 - 用戶端配置簡介, 151
 - 目錄檔, 163
 - 回收, 164
 - 回收指令, 169
 - 回收器用戶端配置, 167
 - 回收器伺服器配置, 166
 - 回收器排程, 186
 - 安裝, 152
 - 伺服器配置檔案, 158
 - 伺服器配置簡介, 150
 - 技術簡介, 150
 - 限制, 149
 - 配置, 152
 - 配置回收, 167
 - 配置回收 (方法 2), 188
 - 配置範例, 152
 - 啓用歸檔, 161
 - 程式庫目錄檔, 151
 - 虛擬裝置, 151
 - 需求, 149
 - 編輯 archiver.cmd 檔案以配置回收, 168
 - 編輯 mcf 檔案, 155
 - 編輯 recycler.cmd 檔案, 170
 - 歸檔, 152
 - 簡介, 148
- sam-robotsd 常駐程式, 3
- sam-serverd 常駐程式, 151
- samset(1M) 指令
 - 啓用裝置記錄檔, 196
- sam-sonyd 常駐程式, 3
- sam-stkd 常駐程式, 3
- samu(1M)
 - arrun 指令, 187
 - R 畫面, 163
 - s 畫面, 162
 - v 畫面, 164
- scrgadm 指令, 191
- SEF, 200
 - sysevent 處理程式, 204
 - 記錄檔, 203
 - 報告輸出, 201
- sefdata 檔案, 200
- sefreport(1M) 指令, 200
 - 選項, 201
- segment(1) 指令, 199
- showqueue(1M) 指令, 34
- showrev(1M) 指令, 154
- Sony 8400 PetaSite 自動程式庫
 - 作業, 211
- Sony 網路連結式自動程式庫
 - 作業, 214
- ssum -e, 資料驗證, 41
- stage 歸檔器指令, 61
- stager.cmd 檔案, 119
 - bufsize 指令, 121
 - drives 指令, 120
 - logfile 指令, 122
 - maxactive 指令, 125
 - 建立, 120
 - 範例, 125
- startage 歸檔器指令, 78

- startcount 歸檔器指令, 78
- startsize 歸檔器指令, 78
- Sun Cluster
 - SAM 配置, 4
 - 高可用性 SAM, 189
- Sun StorageTek 5800, 磁碟歸檔, 84, 86
- Sun StorageTek ACSLS 連結式自動程式庫作業, 215
- Sun StorageTek 程式庫, 使用 HA-SAM, 194
- SUNWhasam RTR 檔案, 191
- sysevent 功能, 204
- syseventadm(1M) 指令, 204
- syseventd 檔案, 204
- tapealert(1M)
 - 與 SEF, 200
- tapenonstop 歸檔器指令, 73
- tplabel(1M) 指令, 15, 185, 187, 188, 198
- user 歸檔器指令, 59
- VSN
 - pool 指令, 81
 - 池, 範例, 100
 - 使用常規表示式, 80
 - 最低回收增益, 139
 - 關聯指令, 79
- vsn_priority preview.cmd 指令, 127
- vsnpools 歸檔器指令, 81
- wait 歸檔器指令, 54
- weight_age 釋放器指令, 110
- weight_age_access 釋放器指令, 110
- weight_age_modify 釋放器指令, 110
- weight_age_residence 釋放器指令, 110
- weight_size 釋放器指令, 111
- wm_priority preview.cmd 檔案指令, 127

四畫

- 中介資料副本, 66
- 分段備份程式
 - 分段備份中的歸檔角色, 125
 - 分段備份請求的數目設定, 125
 - 定義的, 119

- 指令, 119
- 指定磁碟機數量, 120
- 記錄作業, 122
- 記錄檔案欄位, 124
- 強制分段備份程式請求, 130
- 設定分段備份緩衝區大小, 121
- 部分分段備份, 106
- 預覽佇列, 126
- 錯誤程序, 2
- 簡介, 2, 119
- 文件, xxii

五畫

- 加密, 使用磁帶機, 29
- 卡匣
 - 卸載, 15, 30
 - 清理, 18
 - 移除, 23
 - 匯入, 26, 27
 - 匯入與匯出, 24
 - 匯出, 27, 28
 - 載入, 14, 30
 - 製作標籤, 15
- 可用空間, 已定義, 133
- 可移除式媒體
 - 停止, 13
 - 啟動, 12
 - 檔案, 197
- 目前資料, 已定義, 133
- 目錄, 簡介, 24

六畫

- 回收指令, 71
- 回收器
 - crontab 項目, 141, 144
 - recycler.cmd 檔案範例, 138
 - recycler.sh 程序檔, 142
 - 方法, 135
 - 用於保留歸檔備份的 nrecycler, 145
 - 回復歸檔磁碟歸檔, 85

- 作業理論, 134
- 防止回收, 136
- 忽略程式庫, 139
- 指令, 135
- 指定自動程式庫的回收, 136
- 指定記錄檔, 135
 - 配置, 134
 - 配置用於磁碟歸檔, 143
 - 高標記指令, 139
 - 強制使用 chmed(1M) 指令, 139
 - 最低 VSN 增益指令, 139
 - 郵件通知選項, 139
 - 磁碟歸檔備份, 134
 - 編輯 archiver.cmd 檔案以進行磁碟歸檔, 143
 - 簡介, 2, 133
- 回收器記錄檔, 172, 176, 184, 188
 - no-data VSN, 184
 - partially full VSN, 186
- 有人照料指令, 26
- 自動清理, 21
- 自動程式庫
 - ADIC/Grau, 206
 - Fujitsu LMF, 207
 - IBM 3494, 210
 - IBM 3584, 209
 - SCSI 連結式, 請參閱自動程式庫, 直接連結式
 - Sony 8400 PetaSite, 211
 - Sony 網路連結式, 214
 - Sun StorageTek ACSLS 連結式, 215
 - 目錄檔, 24
 - 作業, 24
 - 供應商指定程序, 205
 - 定義的, 11
 - 直接連結式, 3
 - 指定回收參數, 136
 - 記錄器, 25
 - 常駐程式, 3
 - 清理, 18
 - 開啓, 13
 - 匯入與匯出, 26
 - 網路連結式, 3
 - 稽核, 17
 - 關閉, 14
- 自動裝置, 請參閱自動程式庫

七畫

- 低標記
 - preview.cmd 檔案指令, 127
- 技術支援, xxiv
- 系統錯誤功能, 請參閱 SEF

八畫

- 使用 SAM-Remote 歸檔, 152, 161
- 使用 Sun SAM-Remote 進行回收, 164
- 卸載媒體, 15
- 取消歸檔, 65, 73

九畫

- 信箱, 26
- 保留歸檔備份, sam-nrecycler, 145
- 指令
 - 歸檔, 45
- 重新歸檔, 已定義, 134

十畫

- 容量, 已定義, 133
- 記錄器, 25
- 記錄檔
 - 分段備份程式, 122
 - 回收器, 172, 176, 184, 188
 - 回收器錯誤訊息, 140, 144
 - 啓用裝置記錄, 196
 - 備份, 51, 90
 - 裝置記錄, 195
 - 管理 SEF 記錄檔, 203
 - 請參閱記錄檔, 200
 - 歸檔器, 40, 51
 - 釋放器, 113
- 記錄檔回收指令, 135
- 記錄檔感應頁, 200
- 訊息檔案, 140, 144
- 高可用性 SAM, 4, 189
- 高標記, 104
 - preview.cmd 檔案指令, 127
 - 具有回收器, 139

十一畫

- 區段檔案, 199
 - 歸檔, 199
- 常規表示式, 59
- 常駐程式
 - sam-amld, 126
 - sam-archiverd, 36
 - samarchiverd, 32
 - sam-genericd, 3
 - sam-ibm3494d, 3
 - sam-robotsd, 3
 - sam-serverd, 151
 - sam-sonyd, 3
 - sam-stkd, 3
 - 自動程式庫常駐程式, 3
- 授權
 - 一般資訊, xxiv
- 條碼
 - 清理卡匣, 18
- 清理卡匣, 18
- 清理週期, 重設, 20
- 清理磁帶機, 18
- 處理分段備份請求時發生錯誤, 2
- 軟體
 - 文件, xxii
- 連續歸檔, 49
- 部分釋放, 103
 - 使用者選項, 108
 - 簡介, 106

十二畫

- 媒體
 - 卸載, 15
 - 移動, 24
 - 程式庫, 請參閱自動程式庫
 - 載入, 14
 - 錯誤, 22
- 程式庫目錄檔
 - 檢視, 30
 - 簡介, 24
- 程式庫回收指令, 136
- 程式庫記錄器, 25

十三畫

- 傾印檔案, 使用 sam-nrecycler, 145
- 匯入媒體, 簡介, 24
- 匯出媒體, 簡介, 24
- 裝置記錄, 195
 - 何時使用, 196
 - 事件, 196
 - 啟用, 196
- 資料驗證, 41
- 資料驗證的總和檢查, 41
- 載入通知, 啟用, 28
- 載入媒體, 14
 - 手動載入磁碟機, 30
- 過期資料, 已定義, 133
- 預設
 - 歸檔器, 32
- 預覽佇列, 90
- 預覽請求
 - VSN 決定優先權, 127
 - 依高標記和低標記確定優先權, 128
 - 計算優先權, 129
 - 時間決定優先權, 127
 - 配置範例, 130
 - 排定優先權, 126
 - 規劃, 129
 - 標記指令, 127
- 預覽請求的 VSN 優先權, 127

十四畫

- 磁光裝置, 請參閱自動程式庫
- 磁帶清理設定, 21
- 磁帶儲存裝置, 請參閱自動程式庫
- 磁碟快取記憶體
 - 釋放優先權, 2
- 磁碟區溢位
 - ovflmin 歸檔器指令, 52
 - 範例, 52
 - 檔案, 197
- 磁碟機, 具備加密功能, 29
- 磁碟機, 清理, 18

- 磁碟歸檔, 82
 - 在 Sun StorageTek 5800 上, 84
 - 指令, 84
 - 配置, 83
 - 配置回收, 83, 134, 143
 - 啓用, 85
 - 範例, 87
- 製作卡匣標籤, 15

十五畫

- 標記預覽請求指令, 127

- 稽核

 - 自動程式庫, 17
 - 磁碟區, 17

- 請求檔案, 請參閱可移除式媒體檔案

十六畫

- 獨立式磁碟機

 - 載入媒體, 30

- 錯誤訊息, 針對回收器, 140, 144

十七畫

- 檔案, 設定屬性, 61

- 檔案系統

 - 簡介, 1

十八畫

- 歸檔作業的逾時, 38

- 歸檔的佇列時間通知, 39

- 歸檔記錄

 - 備份, 90

- 歸檔組

 - 名稱, 56
 - 搜尋條件, 56
 - 路徑, 56
 - 複製數目, 63
 - 檔案屬性, 56

- 歸檔請求, 36

 - 排程, 37

- 歸檔請求檔案, 34

- 歸檔器

 - allsets 歸檔組, 67

 - archmax 參數, 37

 - drivemin 參數, 37

 - drives 參數, 37

 - fillvsns 參數, 37

 - join 參數, 36

 - ovflmin 參數, 38

 - sort 和 -rsort 參數, 36

 - VSN 池, 81

 - VSN 關聯指令, 79

 - 分段備份中的角色, 125

 - 在 archiver.cmd 中指定檔案系統, 55

 - 佇列時間通知, 39

 - 作業理論, 31

 - 作業簡介, 33

 - 決定歸檔時間, 73

 - 使用常規表示式, 59

 - 定義的, 31

 - 延遲歸檔器啓動, 54

 - 阻止歸檔, 57

 - 保留參數, 36

 - 指引, 90

 - 指令, 43, 45

 - 指定檔案系統資料的備份, 66

 - 指定歸檔記錄檔案, 51

 - 指定歸檔間隔, 50

 - 指定歸檔緩衝鎖定, 70

 - 指定歸檔請求的磁碟機數目, 68

 - 重新命名事件通知程序檔, 51

 - 記錄檔, 40

 - 記錄檔範例, 40

 - 區段檔案, 199

 - 常駐程式, 32

 - 強制歸檔請求, 130

 - 控制使用的磁碟機數量, 49

 - 控制取消歸檔, 73

 - 控制磁碟區溢位, 52

 - 控制檔案大小, 47

 - 控制歸檔掃描, 49

 - 控制歸檔檔案的大小, 67

- 掃描歸檔, 35
- 排定歸檔請求, 37
- 啓用磁碟歸檔, 85
- 設定自動取消歸檔, 65
- 設定歸檔時間, 65
- 設定歸檔器緩衝區大小, 48, 68
- 設定屬性, 77
- 連續歸檔, 35, 49
- 備份定義指令, 63
- 資料驗證, 41
- 逾時, 38
- 預留的 VSN, 75
- 預留磁碟區, 74
- 預設, 32
- 預覽佇列, 90, 126
- 磁碟歸檔, 82
- 磁碟歸檔指令, 84
- 磁碟歸檔配置, 83
- 範例, 91
- 辨識要歸檔的檔案, 34
- 歸檔一個歸檔請求, 38
- 歸檔時間定義, 32
- 歸檔排程, 78
- 歸檔組, 32
- 歸檔組成員身份, 56
- 歸檔組成員身份衝突, 62
- 歸檔組參數, 36
- 歸檔組處理指令, 66
- 歸檔媒體定義, 1, 31
- 歸檔間隔定義, 32
- 歸檔請求, 36

- 歸檔優先權, 33
- 簡介, 1
- 關聯歸檔, 72

十九畫

- 關聯歸檔, 72

二十畫

- 釋放與回復釋放指令, 同時使用, 65

釋放器

- archiver.cmd 檔案角色, 115
- fs 指令, 112
- 手動作業, 117
- 作業理論, 104
- 指令, 109
- 指令檔案, 109
- 候選者定義, 105
- 記錄檔, 113
- 配置, 116
- 部分釋放, 103, 105
- 部分釋放, 使用者選項, 108
- 部分釋放選項, 107
- 優先權, 105
- 檔案時間, 105
- 簡介, 2, 103, 104
- 釋放優先權指令, 109
- 權數, 105

